Holographic Volumetric 3D Displays

Javid Khan

Submitted for the degree of Engineering Doctorate

Heriot Watt University

School of Engineering and Physical Sciences

May 2014

Abstract

The development of a true three-dimensional display that can recreate the light field of an object or scene has been a research goal over the past century. This has recently intensified with the increase in content and proliferation of 3D data across many fields of science and engineering. The literature says that the best approaches to the problem includes holography. The research community has been busy developing the ideal holographic display. However, there are considerable technological challenges that must be overcome before this is a viable proposition.

The solutions proposed in this thesis turn the problem around and take a bottom-up view, rather than the traditional top-down one. It turns out that placing certain constraints on the image enables practical implementations of holographic displays. The approach begins with low information content building up towards higher resolution displays by exploring various techniques aimed at improving performance. The work starts with empirical experimental methods that lead to a framework with theory, numerical methods, simulations and tools to design displays. The results are large, bright displays viewable under ambient conditions with human interaction. The work opens up a new avenue of research that lies between volumetric technology and contemporary holographic display research.

This thesis is dedicated first and foremost to my loving family, my wife Wendy and my children Simon and Jasmine. They have endured an uprooting, changes of jobs, schools and country so that they could not only share in this journey but support me by celebrating the victories and standing by the inevitable disappointments.

I also dedicate this work to my friends, wider family and early investors of Holoxica Ltd; especially Michael Castle, David Ellard and Rohitha de Silva for supporting the creation of a risky high-tech venture.

To all those high-tech entrepreneurs for their vision and persistence, for thinking outside the box, whilst audaciously challenging convention.

Acknowledgements

This Engineering Doctorate thesis has involved numerous collaborators from universities, research institutes, companies and individuals. Contributions have ranged from the provision of expertise, resources, funding, equipment or components to making measurements. These contributions are explicitly cited and acknowledged throughout the document. All of the designs, theory, models, simulations, code and algorithms presented in this thesis were conducted by myself. Unless otherwise acknowledged, all of the work presented in this thesis is my own original work.

First and foremost, I would like to acknowledge my supervisors Prof. Ian Underwood, the industrial supervisor and Prof. Alan Greenaway, the academic supervisor. I thank Ian for his business knowledge, entrepreneurial advice and for keeping me on the straight and narrow path. I had many long and fruitful discussions with Alan whose technical advice and insights were invaluable. Both supervisors helped me navigate the interfaces between the scientific/academic and the engineering/industrial worlds, which is not always straightforward. Finally, I would acknowledge Prof. Andy Harvey, who was the director of the Photonics Industrial Doctorate Centre, for accepting such an unconventional project. After all, it is exceptionally rare for someone to turn up at a university, ask to do some research and to be accepted for such an extraordinary proposal. Ian, Alan and Andy understood the risks and pitfalls associated with such a technology venture and I thank them for their foresight and patience. I sincerely hope that it has all been worthwhile from their perspective also.

This Engineering Doctorate is supported by EPSRC with matching funds from Holoxica Ltd, the industrial sponsor. Certain parts of the work were supported by the TSB (Technology Strategy Board). Further smaller grants were received from Scottish Enterprise, SUPA (Scottish Universities Physics Alliance) and TTOM (Technology Transfer Opportunity Mechanism) by the Scottish Optoelectronics Association. These grants are also acknowledged throughout the document.

Table of Contents

List of Figures

Glossary

AEngD	Association of Engineering Doctorates
C/C++	A compiled computer programming language
CCD	Charge-Coupled Detector
CT	Computed Tomography imaging with X-rays
CGH	Computer Generated Holography
DCG	Di-Chromated Gelatin
DLP	Digital Light Processor
DMD	Digital Micro-mirror Device
DOE	Diffractive Optical Element
EPSRC	Engineering and Physical Sciences Research Council
FFT	Fast Fourier Transform
FOURIOT	FOUrier Optical design Tool
GPU	Graphical Processor Unit
HOE	Holographic Optical Element
H1/H2	Two-step holography with a master transmission copied on to a reflection
IDE	Integrated Development Environment
IHMA	International Hologram Manufacturers Association
ITFT	ITerative Fourier Transform algorithms
LCoS	Liquid Crystal on Silicon
Matlab	An interpreted Mathematical computer programming language
MEMs	Micro Electro-mechanical Machines
MRI	Magnetic Resonance Imaging
PDLC	Polymer Dispersed Liquid Crystal
PECVD	Plasma Enhanced Chemical Vapour Deposition
PP	Photopolymer photosensitive material
PyGame	A gaming library for Python
Python	An interpreted object-oriented computer programming language
RGB	Red Green and Blue
SLM	Spatial Light Modulator
SMC	Scottish Microelectronics Centre
SNR	Signal to Noise Ratio
SPIE	International Photonics Society
SUPA	Scottish Universities Physics Alliance

TTOM	Technology Transfer Opportunity Mechanism
TSB	Technology Strategy Board
VGA	Video Graphics Array
WXGA	Wide eXtended Graphics Array

Publications and Dissemination

Papers and patent

J. Khan, *Three dimensional holographic volumetric display*, US Patent **8625183** (2014).

J. Khan, I. Underwood, A. Greenaway, and M. Halonen, *A low-resolution 3D holographic volumetric display*, Proc. SPIE, **7723**, Optics, Photonics, and Digital Technologies for Multimedia Applications, P. Schelkens, T. Ebrahimi, G. Cristobal, F. Truchetet, and P. Saarikko, Eds., p. 77231B–7, SPIE, Brussels, Belgium (2010) [doi:10.1117/12.858596].

J. Khan, C. Can, A. Greenaway, and I. Underwood, *A real-space interactive holographic display based on a large-aperture HOE*, Proc. SPIE, **8644**, Practical Holography XXVII: Materials and Applications, p. 86440M–86440M, SPIE, San Francisco (2013) [doi:10.1117/12.2021633].

Posters and oral conference presentations

J. Khan, *Low Resolution 3D Holographic Displays*, poster presented at the EngD Conference Edinburgh, Jul 2009

J. Khan, *Holographic Volumetric 3D Displays,* poster presented at the Institute of Integrated Systems Workshop, Edinburgh, Jan 2010

J. Khan, I. Underwood, *A pico-projector driven 3D Holographic Volumetric Display,* presented at The SID Personal Projection and Information Displays Conference, Dresden, Mar 2010

J. Khan, *Animated Holographic 3D Displays*, presented at the EngD Conference Edinburgh, Jul 2011

J. Khan, *3D Holographic Visualisation*, presented at The Photonex Conference, Coventry, Oct 2011

J. Khan, *Animated Holographic 3D Displays*, presented at The HoloPrint-HoloPack IHMA Conference, Las Vegas, Nov 2011

J. Khan, *Fixed Frame 3D Holographic Displays*, presented at SPIE Photonics West Conference, San Francisco, Holography Group Meeting Jan 2012

J. Khan, K. Nguyen, I. Underwood, *New Applications of Holographic Projection in 3D Image Capture and Display*, presented at 4th Workshop on Liquid Crystals for Photonics 2012, 9-11 December, 2012, Hong Kong

J. Khan, *3D Holographic Imaging of Scientific, Molecular and Medical Datasets*, presented at Sinapse Annual Scientific Meeting, Aberdeen, May 2013

J. Khan, *Interactive Holographic HUD-style Display*, poster presented at SID Mid-Europe Conference Gent, Apr 2013

Press and trade news articles

I. Lancaster, *Holoxica's Dynamic Hologram Display*, in Holography News, IHMA, 25, pp. 1–2 (2011).

D. Jeff, *The 3D Force may be with Holoxica*, Scotland on Sunday newspaper, Jan 2012

C. Chinnock, *Holoxica Debuts Dynamic Hologram Display*, in Large Display Report 10, pp. 42–43 (2012).

C. Chinnock, *Holoxica Demonstrates Second Generation HOE-based 3D Display*, 3D News Report, Mar 2013

I. Lancaster, *Real-time 'Thin Air' Display from Holoxica*, in Holography News, IHMA, 27, pp. 1,4 (2013).

J. Bradbury, *'The Gadget Show' featuring HUD-style Holographic Display,* TV Broadcast, Channel 5 (UK), Mar 2013

J. Khan, *Biomedical Holographic 3D Imaging*, Laser Focus World, Jul 2013

Prizes and awards

Edinburgh Entrepreneurs Pitching Contest winner, Santander Bank, Mar 2010

Scottish Technology Prize, Thales, Oct 2010

Photonics Scholarship Award, SPIE, Jan 2012

Best HOE Application Award, IHMA, Nov 2013

Engineering Writer of the Year Award, AEngD, Nov 2013

Chapter 1 - Introduction

Electronic displays are ubiquitous and widespread but they are currently constrained to two dimensions. The human visual system is highly tuned to three dimensional perception which has led to an increasing demand for 3D displays. However, the transition to three dimensions is not easy and is presently restricted to simplistic stereoscopic illusions. It remains to be seen whether these stereo illusions will become widely adopted but history has shown that stereoscopic displays can be popular for a brief time before dying out [1,2].

This chapter presents an overview of 3D visual perception, principles of 3D and the current state of that art in 3D displays. The emphasis is on those technologies that appear to be most promising from a technical, practical and commercial perspective [3].

1.1 Human 3D Perception

The human visual system perceives the 3D world as a pair of two-dimensional images projected on to the back of the retina through the pupils of the eyes. These 2D images are then converted back into 3D within the brain so it is important to understand how these processes work. The brain processes a combination of binocular and monocular information in order to recreate the 3D scene.

Stereopsis is responsible for binocular vision due to the lateral displacement of the eyes, where a pair of offset 2D images are used to synthesise depth information from the scene. Vergence is the other key binocular cue for depth perception, which is the ability of both eyes to triangulate and focus on an object where the eyes converge on objects nearby or diverge on objects further away.

Although binocular depth cues are the dominant means of 3D perception, there are a surprising number of monocular visual cues that enhance the experience [4]:

- Accommodation
- Motion parallax
- Linear perspective
- Relative size
- Occlusion
- Texture, lighting and shading

Accommodation is the ability of the eye to focus on objects at different distances by changing the focal length of the lens on the retina via the muscles. Stretching the lens

increases the focal length and flattening causes it to be decreased. The eye can normally accommodate objects from about 7 cm to infinity in about 350 ms. Vergence and accommodation go hand-in-hand and any conflicts between the two lead to problems with 3D perception [5,6], which is the case with several conventional display technologies as discussed in the next section. Motion parallax is another related depth cue associated with the motion of the viewer across a scene where objects in the foreground appear to move more quickly than those in the background. Relative size is where the magnitude of an object is perceived to be related to its distance. Since the brain is familiar with many objects from experience, nearby objects appear to be larger than objects in the distance. Linear perspective is a geometric effect where parallel lines appear to triangulate towards a distant vanishing point located on the horizon. Occlusion of objects relative to others provides depth cues about the position of objects where backgrounds objects tend to be hidden or obscured by foreground objects. Finally, lighting, shading and texturing all enhance the perception of depth within a scene. Indeed, all of these factors are used to great effect in conveying 3D information in 2D imagery and artwork. There are a number of other factors that are important for 2D as well as 3D displays including persistence of vision, colour perception etc. and these are tackled in other texts [7,8].

1.2 Overview of 3D Display principles

An ideal 3D display needs to be able to recreate the depth cues discussed in the previous section in order for the human brain to perceive the scene correctly. The technologies employed within 3D displays can be broadly defined across the categories listed below.

- Stereoscopic Displays
- Auto-stereoscopic Displays
- Head-mounted Displays
- Volumetric Displays
- Holographic Displays

Currently, the most popular technologies are based on variants of LCD screens with stereoscopic glasses or lenticular arrays. Prolonged viewing of such stereoscopic displays leads to discomfort and disorientation such that manufacturers have started to issue health warnings with the first crop of 3D-TV products. Wearing glasses is inconvenient and uncomfortable whereas lenticulars limit the number of viewers. Head-

mounted and volumetric displays have been around for a long time but they have never really caught on, largely because of inconvenience, technical/mechanical and cost issues. Holographic displays are currently state of the art and are difficult to make.

According to recent review papers [9] and scientific literature [10], the best 3D displays are likely to be based on technologies that can recreate the light field of the 3D scene and hence retain all of the depth cues discussed in section 1.1. The only technologies currently known to reproduce the physical light field distribution are based on the principles of integral imaging via refractive micro-optical arrays or holography based on diffractive optics.

The following sections outline a review of the most popular and emerging 3D display technologies with commercial potential. The coverage is by no means exhaustive and only reflects a small subset of the technological spectrum.

1.2.1 Stereoscopic displays: glasses

Stereoscopic glasses have been around for many years in various forms starting with anaglyph techniques popularised by 1950's cinema. Currently, the most popular technologies employ simple polarised glasses (Figure 1) which are worn for cinema and television. Cinemas are equipped with special projectors that simultaneously project polarised images on to a large polarisation-preserving screen. Polarising glasses, however, have inferior 3D performance to other systems as there are problems with image crosstalk effects or ghosting. The typical contrast ratio is 100:1. The main advantage is cost which can be as low as a few dollars.

Figure 1: Polarising glasses, © RealD

Colour filter glasses have come a long way since the early simple red and cyan filters of the 1950's. Current state of the art filter glasses employ multicolour filtering using narrowband notching filters across the visible spectrum with separate RGB filters for each eye. The screen is a specially adapted LCD or a pair of projectors with six colour filters centred on each of the sub-bands where both images are presented simultaneously. As the notch filtering can be fabricated to a high degree of optical precision, the contrast ratio is high at 10,000:1. The main current disadvantage is cost as this technology is relatively new. The glasses cost a few hundred dollars. This multicolour filtering technique does

Figure 2: Active shutter glasses, © Sony

not suffer from the flickering or ghosting problems associated with shutter or polarising glasses.

Active glasses (Figure 2) use liquid crystal shutters that are synchronised with an LCD or projection display. The liquid crystal shutter consists of a polariser and a nematic liquid crystal layer that blocks the light through the exit polariser when a voltage is applied. The mechanism masks out each eye in turn and high refresh rates are needed (~200 Hz) to prevent flickering. Synchronisation is either through a wired interface to a computer or wirelessly via infra red or radio. The main application is computer gaming. The typical contrast ratio is about 500:1. The devices tend to be bulky as they contain electronics as well as a power source for wireless versions. The cost is around $100.

Stereo glasses are often used by seated viewers so motion parallax is not a great issue. The eyes are focused on the plane of the screen but the 3D image is presented outside this plane, which leads to a disparity between vergence and accommodation. This can only be tolerated for short periods of time by untrained viewers who experience discomfort or headaches with this change of focus.

1.2.2 Autostereo displays: lenticular and parallax barrier

Auto-stereoscopic displays do not require any eyewear for viewing, unlike the glasses discussed previously. These are are similar to glasses-based displays because they use a pair of stereoscopic images that are presented to both eyes within a spatially constrained 'sweet spot'. Such displays are commonly used in commercial applications such as advertising as well as television.

A lenticular lens consists of an array of cylindrical lenslets, placed in front of the columns of conventional LCD. The 3D image is formed by interleaving the left and right images of a stereoscopic view from a scene on alternate columns of the LCD (Figure 3, lower part). For viewing, the lenticular array ensures that the interlaced images from alternate columns is viewed correctly by each eye. The parallax barrier is a simpler configuration [11] where a mask is placed in front of the interleaved screen whereby the respective left/right views are obscured by the barrier for each eye (Figure 3, upper part). Some products such as the Nintendo 3DS handheld gaming console employs such a parallax barrier using a secondary liquid crystal layer that can be adjusted to switch between 2D and 3D mode.

Auto-stereoscopic displays only have a limited 'sweet spot' with no parallax which means that they can only be viewed by one person at a time in one position. It is possible to have several sweet spots such that they can be viewed by multiple persons.

However, each sweet spot acts to halve the resolution of the 3D image so a two Megapixel (high definition) LCD screen would have a 3D resolution of one Megapixel. Two sweet spots would imply a 3D resolution of 0.5 Megapixels and so on. This makes it suitable for viewing from certain positions or when seated at a workstation, for example.

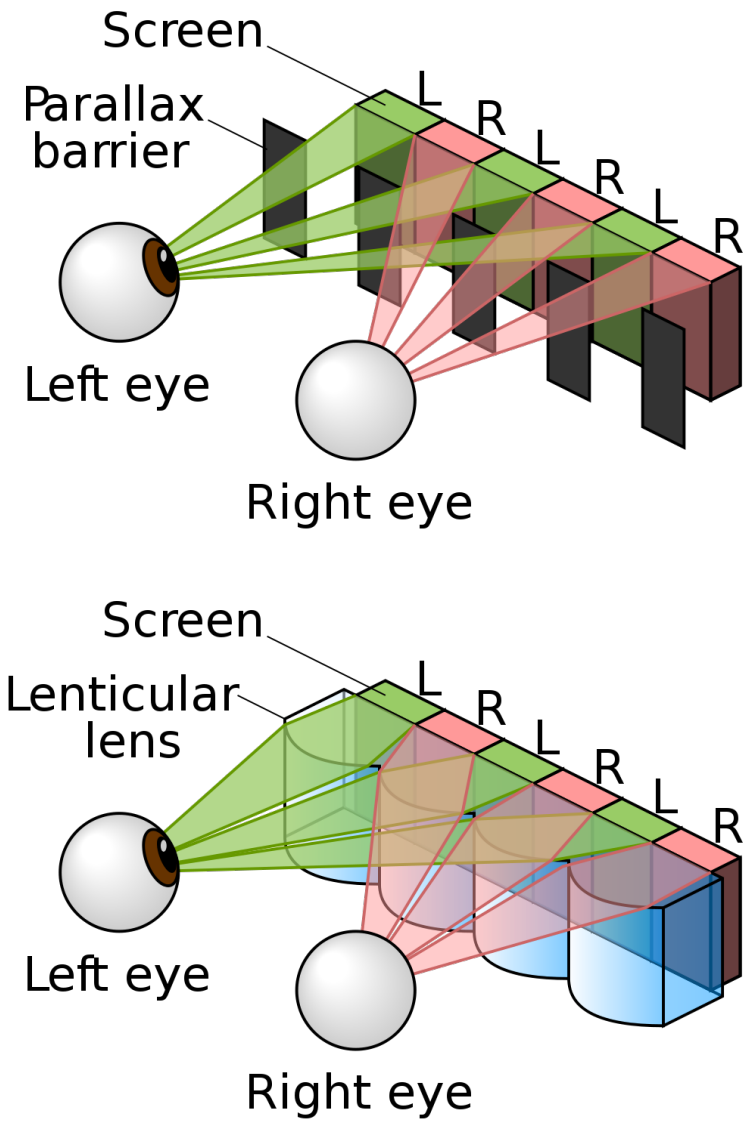

Figure 3: Parallax barrier and lenticular configuration, from [11]

The 3D images cannot be very deep because the viewer's eyes need to focus on the display and not the 3D object which is viewed out of the plane of the display. This leads to the same vergence-accommodation problem as stereoscopic displays. Furthermore, since viewers are likely to move around the sweet spot, this technology does not have motion parallax which further reduces the impact of the 3D effect. Despite the limitations in the 3D experience, this is likely to be popular in the market for applications such as advertising. The displays are relatively cheap to make, only

marginally more expensive than a standard LCD. They are also full colour and high resolution. The current market price is around a thousand dollars.

1.2.3 Volumetric displays

The first 'cubic' or volumetric display was proposed by Baird in 1932 where a 3D surface is encoded as a series of contours that is decoded as a depth map using a mechanical scanning mechanism [12, pp. 138–142]. Volumetric displays belong to a class of three-dimensional display technologies that produce volume-filling imagery [13,14,15, p. 308]. The volume is usually divided into identical volume elements, or voxels, placed on a grid. The arrangement is analogous to pixels in a two dimensional display. The 3D image is constructed by activating any number of voxels. The display is usually controlled by a computer that can make moving images in colour. Volumetric displays are autostereoscopic so viewers do not need any specialised equipment like eyewear or headsets to view a three dimensional image. Furthermore, the image is visible by any number of viewers.

There are many technical approaches to volumetric displays which range from bouncing light beams off smoke or a fine mist or even banks of interconnected LEDs suspended in a volume. Other esoteric approaches light up plasma dots within a gas using intersecting laser beams to stimulate plasma discharges [16], Figure 4, or energising particles in a volume [17]. This section focuses on popular commercial volumetric displays which have been reviewed in the literature [18] and [14,19].

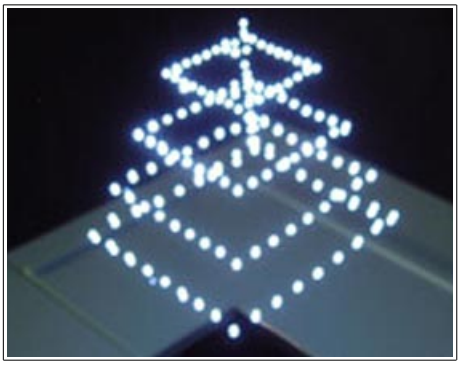

Figure 4: Plasma volumetric display from Keio University, [16]

A common category of volumetric displays includes closed volume display that use the principles of swept volume projection. These use mechanical methods based on arrays of LEDs that spin very quickly around an axis. The LEDs are modulated in full colour, thus presenting the eye with the illusion of a volumetric display. The Sony 3D-360 prototype, Figure 5, comprises a 96x128 colour LED array where the cylinder diameter is 13 cm and the unit is 27 cm high. The 3D image is somewhat blocky, as can be seen from Figure 5 and it is uncertain whether Sony will commercialise this. A similar commercial display is the Kinoton Litefast product which is a full-height (over 2m high) colour spinning display with resolutions around 600x800 pixels (VGA). These units are used in advertising and cost from $45,000 to $90,000.

Figure 5: Sony 3D-360 display prototype

Other swept volume displays are based on moving screens, utilising either translation or rotational motion. Actuality Systems developed the Perspecta display which resembles a crystal ball, depicted in Figure 6; although the company is not currently trading and the product is no longer available. It is based around a DLP microdisplay that projects images onto a round semi-transparent screen that rotates 45 times per second around a vertical axis. The projected information is divided into 198 radially segmented 'slices' that combine to create the illusion of a 3D image. The screen is half transmissive and half reflective which only displays the voxel when it sweeps through that particular point in space. The device creates a full-colour, glowing translucent 3D image. The projector has a resolution of a million pixels which gives a virtual resolution of around 100 million voxels [20,21]. The display exhibits full parallax which means

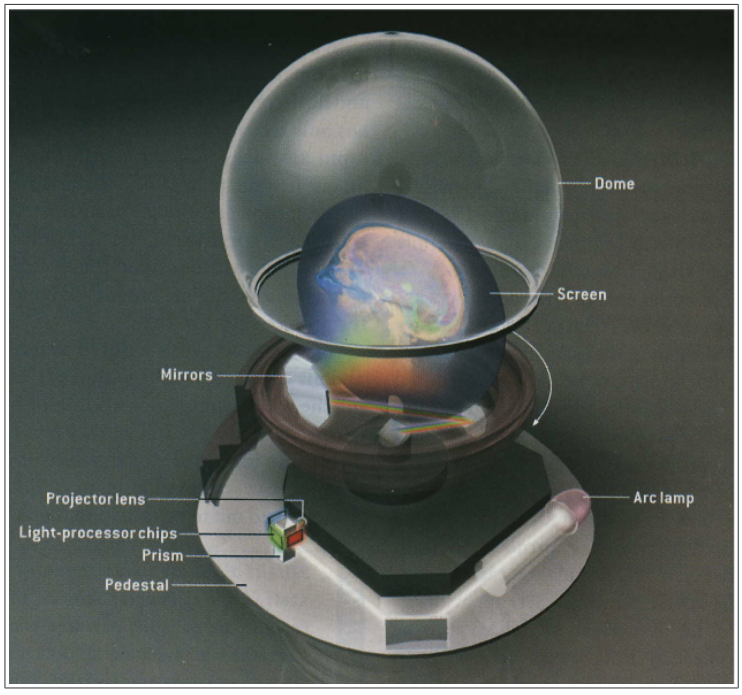

Figure 6: Perspecta Crystal Ball © Actuality Systems

that the viewer can see 360 degrees round the image.

Apart from the cost, these types of swept-volume displays have several drawbacks. Since they are dependent on a high speed mechanical component, their reliability is limited and there are safety concerns about the spinning parts which need to be shielded behind protective glass. It is unclear whether the displays would be suitable for moving images i.e. 3D animation. Such a display is not scalable in terms of size, which means that a larger display would need a larger screen and there are mechanical limits to how fast such a screen could be rotated. On the other hand, scalability in terms of voxel resolution is quite good as this evolves with higher resolution LED displays. The image is a 'virtual' image which means that it needs to be viewed behind the glass and it cannot be projected towards the viewer and the viewer cannot directly interact with it.

The Depthcube product was developed by Lightspace Technologies [22,23]. It looks rather like a 19-inch cathode ray tube monitor measuring 40x30 cm and about 10 cm thick, depicted in Figure 7. The display stacks twenty different liquid crystal shutter layers on top of each other with an image projected successively on each one, which builds up a volumetric 3D image. Each liquid crystal layer can change rapidly from a transparent state to a light-scattering one. The layers are activated in turn by successively switching each one to light-scattering mode and making all the other layers transparent, effectively enabling each 'pixel' on a screen to appear as a voxel. The display is a DLP-based rear-projection system with a different image projected on to each layer fifty times a second. This technology has the advantage that it does not have any mechanical parts. Digital signal and image processing techniques can be used to improve the resolution using interpolation and dithering algorithms. The images are produced in full colour with about 15 million physical voxels.

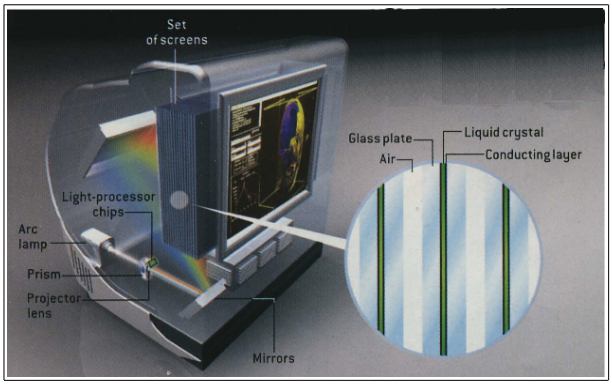

Figure 7: Depthcube display © Lightspace Technologies

The Depthcube display cost around $50,000 in 2009 and it is claimed that, with

volume production, the cost could go down to a few thousand dollars. The main technical drawback is that the 3D images have limited depth resolution due to attenuation through the stack. As more LC layers are added, they serve to attenuate the projected images at the back of the display and this also increases the required refresh rate which is limited to 100-200 Hz for conventional LCDs.

Volumetric displays meet most of the binocular criteria for a 3D display in terms of vergence, accommodation and motion parallax. However, they do not perform well on the other depth cues such as lighting and shading given that the voxels are often translucent. Hence, it is generally not possible to realise occlusions with this technology. This approach is constrained to certain types of applications that deal with stacks of volume data, e.g. medical scans from Magnetic Resonance Imaging (MRI), Computed Tomography (CT) or Ultrasound equipment.

1.2.4 Integral imaging displays

Integral imaging is an alternative 3D formation method to light field generation which is based on ray optics. Integral imaging was first proposed over a hundred years ago by Lippmann [24]. The technique is based on integrating a set of rays emerging from 2D elemental images recorded from slightly different perspectives across the scene. The arrangement consists of a matrix of lenslets or pinholes located just behind a similar arrangement of tiny 2D photos. Each tiny lens views the 3D object from a slightly different perspective than a neighbouring lens, resulting in simultaneous reconstruction from an array of discrete perspectives. This enables the viewer to perceive a 3D representation of the object, depicted in Figure 8. In this figure, the tiny photos are

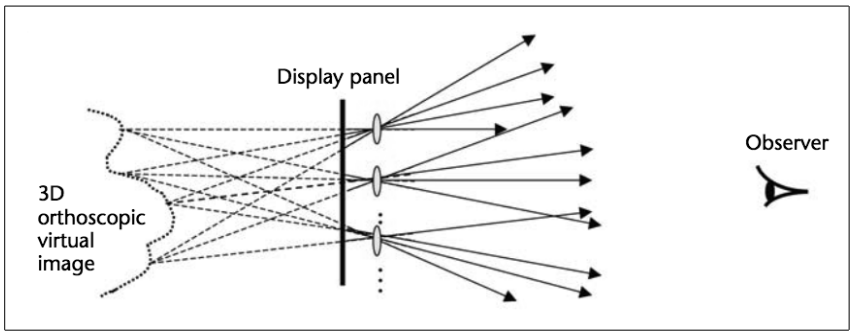

Figure 8: Integral imaging, from [25]

replaced by a display panel which can recreate the array of small images similar to the lenticular LCDs described in section 1.2.2. Indeed, lenticular displays can be understood as a special case of a more generalised integral imaging method. The 3D image resolution is determined by the size of the lenslet matrix whereas the angular resolution

or parallax depends on the lens diameter and pitch as well as the pixel density of the display panel.

Although the idea has been around for a long time, integral imaging has proven to be difficult to implement in practice. Also, capturing the 3D image and transferring it to a display panel is not trivial, involving lenslet arrays and CCDs. Additional optical tricks are required to convert between pseudoscopic and orthoscopic as well as real and virtual images. The issues have been extensively researched in an attempt to improve the performance of the technology using spatial and temporal multiplexing [25].

An increasingly popular approach is computational integral imaging which leverages techniques from 3D computer graphics. The advantage is that the object or scene can be generated completely synthetically from 3D model descriptions by the computer. The rendering of multiple 2D views of the 3D scene can be done using a 'virtual' camera that can pan around the object in any direction. This gives the designer much more freedom than physical or mechanical approaches to making stereograms. In fact, computational integral imaging can be combined with holography whereby the elemental 2D image and lenslet are combined into a “holographic pixel”. A matrix of such pixels can be used to make a computer-generated stereogram by means of a holographic printer, described in section 1.3.3.

The most advanced commercially available 3D display is made by Holografika [26]. The display, known as HoloVizio, can show high resolution horizontal parallax full colour images. It is based on computer generated integral imaging where the light beams that form the 3D image are produced by a set of 2D projectors through a semi-holographic screen. Each point of the semi-holographic screen can emit rays of different colours in different directions under computer control. The screen is a proprietary holographically recorded randomized (non-periodic) surface structure with a specific angular relief profile which is independent of wavelength [27]. For a horizontal parallax configuration, the screen ensures that each incident light beam is diverged by an angle of 0.8^{o}. The light beams generated by the projectors hit the screen at various angles and the screen makes the necessary optical transformations to compose these beams into a continuous 3D view. The 3D image is not associated with any particular set of projectors, it depends on the shape of the object and the geometry of the screen and projectors. This technology can accommodate occlusions within the 3D image, a feature desirable for any general purpose 3D display.

Figure 9: Holovizio display ©Holografika

An example of the display is shown in Figure 9, which is quite large (3.0x2.15x2.7m) and bulky (800 kg) given that it houses an array of up to eighty projectors together with a small computing cluster. The cost of these displays is in the order of several hundred thousand of dollars. However, the cost and bulk will decrease in time with smaller and lower cost micro and pico-projection systems as well as increased computing power.

1.2.5 Holographic displays

The best 3D displays are likely to be based on technologies that can recreate the light field of the 3D scene [28,29]. The only technologies currently known to duplicate the physical light distribution of the scene are based on the principles of holography or integral imaging. Holography is discussed in detail in section 1.3. Integral imaging is based on the principle that a 3D image can be recreated from a series of 2D viewpoints

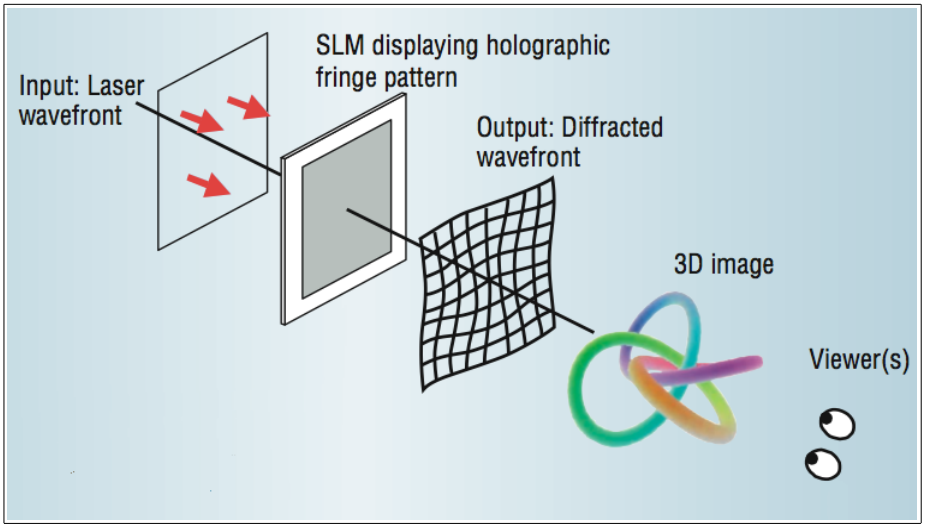

Figure 10: Holographic image reconstruction using SLMs, from [35]

captured around a 3D object or scene. There are no holographic displays on the market today and the quest for the ideal holographic display is still in the research domain.

Computer Generated Holographic (CGH) display research has focused on high resolution 3D imaging based on spatial light modulation (SLM) technology. SLMs are specially constructed micro-displays that provide phase in addition to amplitude modulation, thus providing digitally programmable diffractive holographic elements. The complex diffraction pattern corresponding to the 3D image is first computed and then transferred to the SLM, as depicted in Figure 10, using the principles of Fourier Optics and Computer Generated Holography described in section 1.4.

A series of holographic display prototypes were produced at MIT under Benton, starting with the MIT Mark 1 display from 1989, which is an electro-holographic device base on an acousto-optic SLM capable of producing 1-inch cubic (25x25x25mm) images at 20 frames per second; both monochromatic [30] and colour [31]. The Mark 2 display improves on this in 1994 to achieve a 150x75x75 mm image [32]. The Mark 2 architecture has better optics, faster electronics to scale the earlier system through tiling hardware and increased parallelism [33]. These displays are some of the best known among the research community. The Mark 2 architecture is still in use by researchers at MIT today for trialling new ideas and algorithms for synthesising interference patterns [34].

One of the most advanced CGH display prototypes was made by Qinetiq [35,36] based on arrays of SLMs for military and engineering visualisation applications. This display was a proof of concept and the R&D has unfortunately been discontinued. The cost is unknown but it is likely to be in the order of millions of dollars.

CGH displays require a large amount of computing power to compute the diffraction patterns as well as a considerable amount of bandwidth to distribute the patterns across the SLM array. SLMs need to be based on ultra-high resolution microdisplay technologies with pixel geometries of less than a micron, i.e. gigapixel arrays which are difficult to fabricate reliably. Hence, CGH displays based on SLMs are not currently commercially viable for cost, performance and technology reasons. More practical approaches to CGH displays are described in [37] based on DMDs, and [38] based FLCoS, to make dynamic holographic displays; the latter being full colour at video rates. SeeReal's approach [39,40] uses a LCD display as a simplified SLM together with head tracking to drastically reduce the resolution, bandwidth and computation needed to present a holographic image within the viewer's field of view. Unfortunately, head or eye tracking is a difficult technology to implement in practice and it leads to other problems such as restricting the number of viewers.

The University of Arizona has developed a changeable holographic technology [41] based on a re-settable photorefractive polymeric material that can be recorded optically with laser interference under an applied electric field (5 to 10 kV) and erased under uniform laser illumination; the interference pattern starts to decay as soon as the field is turned off. The first version had monochromatic images that could be erased and rewritten every four minutes. The latest version [42] can rewrite a 10x10 cm hologram in just two seconds using 1 mm^2 holographic pixels. This is achieved with pulsed lasers and colour is achieved through angular multiplexing where three primary-coloured lasers write simultaneously from different angles. This technique is similar to writing stereograms using a holoprinter as described in sections 1.3.3 and 1.6.2. The ultimate goal is to reduce this time to tens of milliseconds for real-time video. A similar technology utilises a holographic thin film based on a proprietary liquid crystal (LC) mixed with a photosensitive material [43] that can be erased and rewritten in just one millisecond. Unlike the Arizona photopolymer, this photo-LC based material is erased and written optically rather than electrically. The photo-LC requires a recording beam at one wavelength whereas the holographic interference occurs at a different wavelength. The interference pattern is maintained as long as the recording beam is active and it is erased once the recording beam is switched off [44]. This technology is potentially suitable for large-sized realtime dynamic full-colour holographic displays. As with the Arizona material, colour is achieved with angular multiplexing of RGB channels.

One of the most advanced emerging 3D displays is the ZScape display by Zebra Imaging, a manufacturer of holographic printing and imaging technologies. The 3D display technology has been developed for and funded by the US military for terrain visualisation purposes amongst others. The display is based on a tiled light-field approach which is somewhere between integral imaging and holography. The optical engine is a complex electro-optical stack, Figure 11, comprising 24 projectors, 6 FPGAs and 3 CPU/GPU nodes [45]. These are arranged as a 3x3 tiled array in for a horizontal format tabletop display.

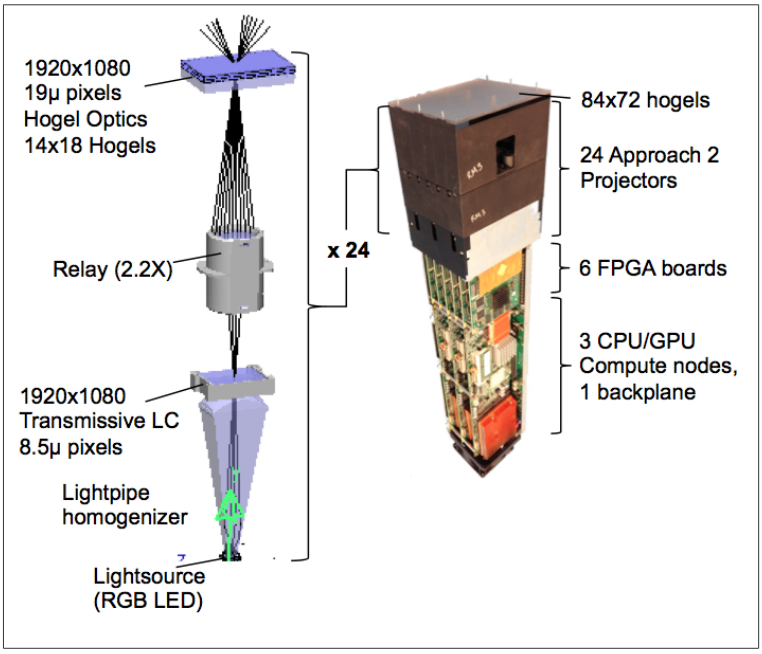

Figure 11: Zebra Imaging optical engine, from [45]

Key performance parameters of the ZScape display include:

- Image size: 0.5m diagonal
- Brightness: 200 cd/m^2
- Contrast: 70:1
- Viewing angle: 90 deg
- Colour: 4000 colours
- Refresh rate: ~3 Hz

The display is rather bulky (150 kg) and power-hungry (2.5 KW). As this is a DARPA-funded project, most of the details are currently classified, including any patents arising.

The research community has been focused on making an ideal 3D display with a top-down engineering approach, which is complex and difficult to implement with current technology. Holographic displays are presently at the basic research stage and require significant advances in many technical fields before they can be commercially feasible [46]. The work in this thesis, on the other hand, focuses on simple holographic displays. In particular, it advocates a simplified bottom-up approach to tackling the complexity of holographic displays with some surprising results.

1.3 Overview of Holography Principles and Techniques

This section presents the basic principles and theoretical foundations of holograms and holography. The term hologram is derived from the Greek meaning 'whole image'. Holograms are able to encode both amplitude and phase information about the 3D object or scene. This information is recorded in such a manner that the light field can be reconstructed to give a true three-dimensional representation corresponding to the original subject. This is in contrast with a photograph, which only stores amplitude information, thus yielding a two-dimensional representation.

Dennis Gabor, a Hungarian scientist working in the UK, presented the foundational concepts of holography in a patent [47] and a series of papers written between 1948 and 1951 [48,49] that were aimed primarily at microscopy. These introduced the notion of storing the 3D information related to a sample as an interference pattern that can subsequently be reconstructed with magnification. Holography remained somewhat obscure due to its dependence on coherent light sources. Things changed radically after the invention of the laser in 1960 when the field had a revival with a huge upsurge in research and development. Lieth and Upatnieks reached a key milestone with the development of the off-axis transmission hologram [50] at the University of Michigan in 1962. At about the same time, Denisyuk pioneered the reflection hologram [51]. The rainbow hologram was invented by Benton in 1969 which opened up new possibilities for viewing holograms under white light [52]. Dennis Gabor was awarded the Physics Nobel Prize in 1971.

Holography covers a diverse range of applications ranging from security on credit cards to areas such as art, design and data storage. There are many different types of hologram and many ways of making them. The most important techniques are discussed in the subsequent sections. Holograms use absorption and or variations in optical thickness to alter the amplitude and phase of a transmitted or reflected wave [53]. A hologram is reconstructed by transmission or reflection of light through the medium. The early holograms were recorded on high resolution photographic film. The modulation is due to interference patterns derived from the interaction of two coherent light waves: a reference beam and a beam reflected off an object. The average fringe spacing or spatial frequency is given by the grating equation:

$$\Lambda = \frac{\sin\theta}{\lambda} \tag{1}$$

where θ is the angle between the object and the film and λ is the wavelength. Λ is

expressed in units of m^{-1}, or more commonly, in lines per mm. Holograms are generally categorised as thick or thin. If the thickness of the medium exceeds the average spatial frequency the hologram is known as a thick hologram, otherwise it is a thin hologram. The distinction can be quantified through a quality metric, Q [54, p. 63]:

$$Q = \frac{2\pi\lambda d}{n\Lambda^2} \tag{2}$$

where d is the thickness of the hologram, Λ is the average fringe spacing and n is the refractive index of the medium. When $Q<1$, it is a thin hologram, otherwise when $Q>20$, the hologram is classified as a thick or volume hologram. Holograms that fall in between these ranges are ambiguous and cannot readily be classified as either thick or thin. Volume holograms may comprise transmission amplitude, transmission phase, reflection amplitude, reflection phase or a combination of these.

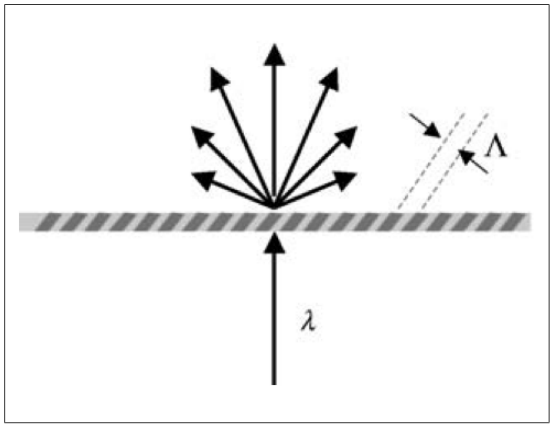

Figure 12: Thin transmission hologram, from [57]

Transmission holograms behave like gratings, requiring coherent illumination through the hologram, based on the principles of diffraction theory. Figure 12 depicts a thin hologram by refractive index modulation.

Reflection holograms can be regarded as tiny wavelength-selective micro-mirrors, requiring ordinary non-coherent illumination on the front surface, based on the principle of Bragg reflection, Figure 13.

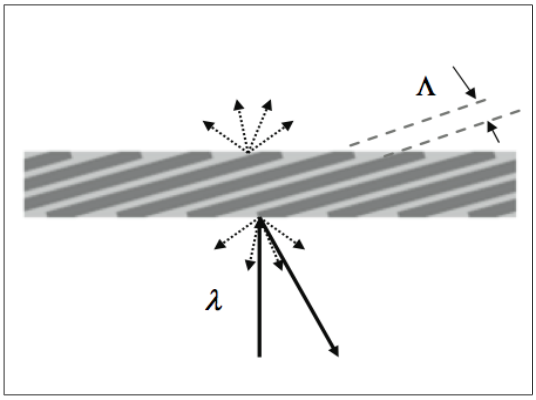

Figure 13: Thick reflection hologram, from [57]

The following sections briefly outline the theory behind thin transmission and thick reflection holograms on film.

1.3.1 Transmission holograms

The transmission hologram is the simplest type of hologram. This is essentially an interference pattern caused by a reference beam and a reflected object beam, which expose the holographic emulsion from the same side. The principle is illustrated in Figure 14 where the object is a cube shape.

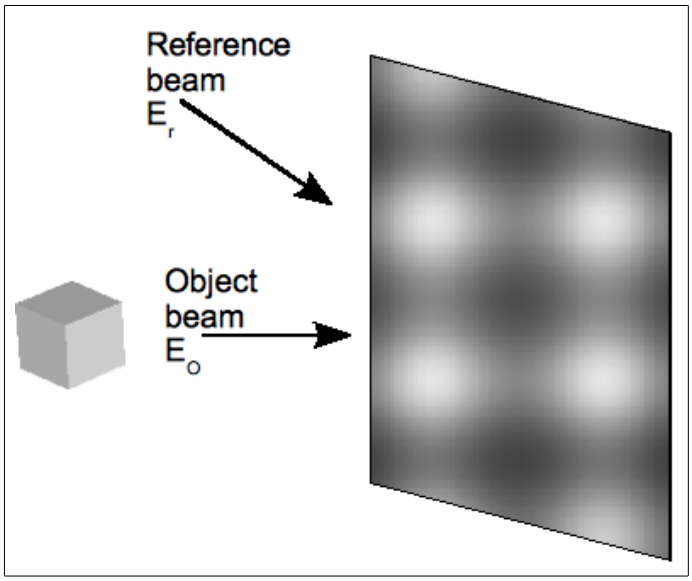

Figure 14: Recording transmission hologram

Consider recording a transmission hologram using a He–Ne laser ($\lambda = 632.8$ nm) with the object and reference beams at an angle of +30° incident on the photographic plate, where the refractive index of unexposed photographic emulsion is about 1.6. The average spatial frequency of the fringes on the recording plate according to equation (1)

is approximately 1,560 lines per mm.

Transmission holograms need to be illuminated by a coherent light source such as a laser. The complex reference and object beams may be written as:

$$E_{Ref} = E_r(x, y)e^{j\omega t} \quad \text{and} \quad E_{Obj} = E_o(x, y)e^{j\omega t} \tag{3}$$

Assuming both beams are mutually coherent and have the same polarisation, then the intensity of the two beams interfering across the surface of the film is given by:

$$I_t(x, y) = |E_{Ref} + E_{Obj}|^2 = (E_{Ref} + E_{Obj})(E^*_{Ref} + E^*_{Obj}) = \alpha + E_o E^*_r + E^*_o E_r \tag{4}$$

The recorded interference pattern contains a constant scalar term α and modulated complex products of the object and reference beams. After the exposed film is processed, the hologram can be written as: $I_h = \beta I_t(x, y)$ where β is a scalar transmission constant.

The initial holograms made by Gabor [48] are known as in-line holograms as they have all of these components in the same plane such that they are all superimposed. This is a somewhat confusing arrangement for viewing. Upatnieks and Leith [50] improved on Gabor's method by placing the reference and object beams at an angle to each other. This means that the image components are separated spatially so the viewing experience is significantly improved. The intensity distribution of such an off-axis hologram is given by:

$$I(x, y) = |E_r|^2 + |E_o|^2 + 2E_r E_o(x, y)\cos(2\pi\Lambda + \phi(x, y)) \tag{5}$$

Where Λ is the average fringe frequency from equation (1), dependent on the angle of separation between the beams. The last term in equation (5) shows that the object amplitude modulates the brightness of the fringes whereas the object phase modulates the distance between the fringes. This equation is revisited and derived in section 8.2.

For viewing, the transmission hologram is illuminated with the original reference beam from the same position to reconstruct the recorded images as shown in Figure 15. Here we have:

$$I_v(x, y,) = \alpha\beta E_r + \beta E_o|E_r|^2 + \beta E^*_o E^2_r \tag{6}$$

The viewer sees an attenuated reference beam together with the reconstructed object (second term) and a modulated conjugate object. The reconstructed object is virtual i.e. the viewer sees the image through the hologram, rather like looking through a window. The conjugate object, on the other hand, is real and is reconstructed outside the plane of the hologram towards the viewer (not shown in Figure 15).

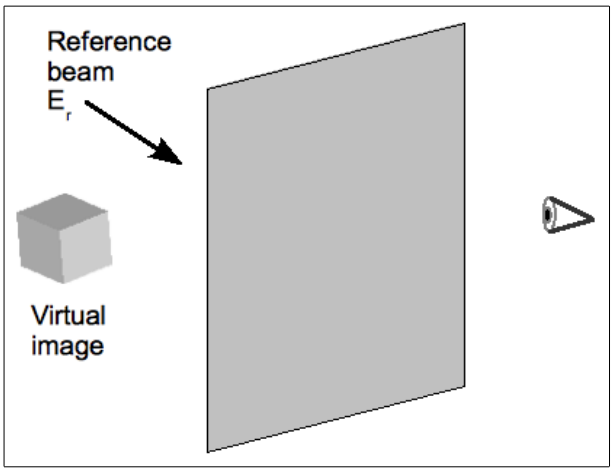

Figure 15: Viewing a transmission hologram

1.3.2 Reflection holograms

Although reflection holograms are more complicated to make than transmission holograms, they are more popular because they are viewable using ordinary light sources rather than lasers. Whereas the transmission hologram is analogous to a diffraction grating across the surface of the film, a reflection hologram can be considered as a series of Bragg mirror layers across the thickness of the emulsion of the film. The reflection hologram is recorded with reference and object beams incident on the emulsion from opposite sides as shown in Figure 16. The beams create a standing wave across the emulsion whose intensity varies at half the laser wavelength. During

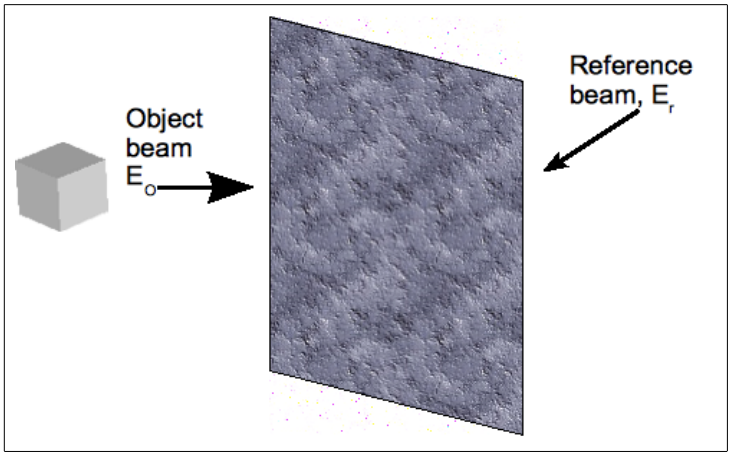

Figure 16: Recording a reflection hologram

exposure, the standing wave envelope is recorded as variations in refractive index across the emulsion. The number of Bragg layers or depth of the hologram is proportional to the intensity of the object whereas the variation in refractive index modulates the phase [55, p. 49].

The Bragg layers act as a reflective filter for the wavelengths that satisfy the Bragg condition:

$$\frac{m\lambda}{2} = \Lambda \sin\theta \tag{7}$$

Where Λ is the inter-layer spacing. Light at twice the fringe spacing is reflected back and constructively interferes with similar reflections from other layers as shown in Figure 17. Only a narrow band of wavelengths is reflected and all other wavelengths pass straight through.

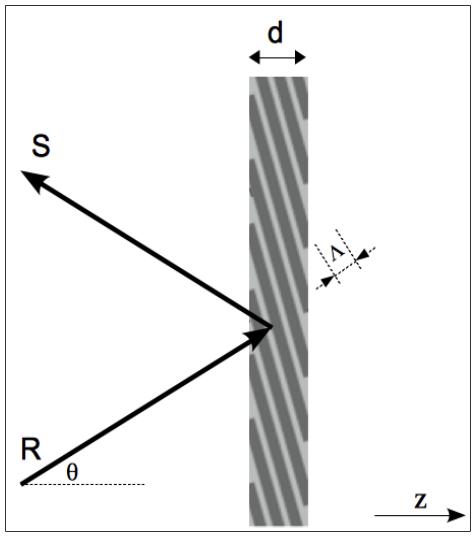

Figure 17: Bragg reflection model

The most widely used model to describe the diffraction efficiency of phase holograms was proposed by Kogelnik [56]. This is a vector theory based on the coupling between a reference wave, *R*, that gives rise to a signal wave, *S*, in a holographic material of thickness d via interaction with the Bragg planes of spatial frequency Λ. The reference wave, *R*, passes through successive Bragg planes and undergoes multiple Fresnel reflections at each interface that cumulatively result in a strong reflection through constructive interference. The reference and signal waves are related through the coupled-wave equations:

$$\frac{\partial R}{\partial z} = -j\kappa \exp(-j\zeta z) S(z)$$
$$\frac{\partial S}{\partial z} = -j\kappa \exp(j\zeta z) R(z) \tag{8}$$

The parameter κ is known as the grating strength parameter and ζ is the detuning parameter used to model the off-Bragg condition [57, p. 193]. Kogelnik's analysis looks at the diffraction efficiency of various thick and thin volume holograms in amplitude and phase configurations. The diffraction efficiency of a thick volume phase reflection hologram is given by:

$$\eta = \tanh^2\left(\frac{\Delta n\, d}{\lambda \cos\theta}\pi\right) = \tanh^2 \Phi \tag{9}$$

Where Δn is the refractive index modulation (typically 0.02 for photopolymer) and d is the thickness of the holographic material (typically 20 μm); assuming λ=500 nm and θ=30 degrees, η≈0.992 or 99.2%. Equation (9) shows that, as Φ exceeds π, the diffraction efficiency η can reach 100%. Practically, the expression shows diffraction efficiency can be improved by increasing the thickness of the holographic material and/or the refractive index modulation. Although thick phase holograms can achieve diffraction efficiencies up to 100%, the situation for amplitude holograms is very different where diffraction efficiencies of just a few per cent can be achieved [58, p. 346].

The system is angularly selective as well as highly frequency selective, which means that it will act as a narrowband reflector from a certain angle. The Kogelnik model is widely used in the analysis of reflection hologram structures, however, it does have some limitations due to a number of underlying assumptions. In particular, the analysis is restricted to a single diffraction order and restricted wavelengths around the Bragg resonance. A modern take on the theory models the Bragg structures as infinite arrays of stacked mirrors by summing the Fresnel reflection at the boundaries [59,60]. This is a more intuitive approach than Kogelnik and the analysis can be generalised to colour and slanted gratings [61].

Reflection holograms have the advantage that they may be viewed using a incoherent point light source. Ordinary bright sources such as a halogen spotlight may be used as illustrated in Figure 18, although latterly LED sources also work well.

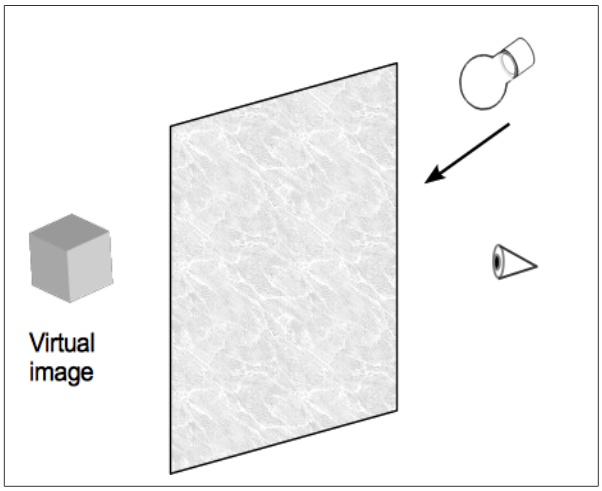

Figure 18: Viewing a reflection hologram

1.3.3 Holographic stereograms and holoprinters

Holographic stereograms combine the principles of holography with integral imaging where a series of 2D photographs of a scene or object from different perspectives are multiplexed on to a hologram. The holographic stereogram was first proposed by DeBitetto in 1969 [62]. The hologram is divided into a series of vertical strips. Each photograph is recorded, in turn, on to adjacent strips on the holographic plate. The result is a smooth 3D representation of the scene or object with horizontal parallax only. Typically, each degree of rotation needs at least three 2D images, so an image viewable over 120 degrees would require 360 different 2D photos. An observer is not able to distinguish individual frames at this angular resolution. Holographic stereograms are also known as multiplexed holograms or multi-channel holograms.

The methods involved in producing stereograms require many repetitive steps requiring a high degree of precision and mechanical stability. This has led to the automation of the process such that stereograms can now be produced in full colour and full parallax using holographic printing technology. A holoprinter is a holographic printing device that can print out full colour digital holograms from a rendered 3D model or a video series. It uses red, green and blue lasers to write a series of dots or holopixels across a holographic medium. Every holopixel contains information about the scene computed from a series of rendered images generated via computer graphics. The holographic medium is typically a polymer or silver halide based film that may

require development after exposure. It is then laminated on to a hard plastic backing. Printing a digital hologram can take several hours as each holopixel dot has to be written individually in three colours, where the colours overlap within the medium. The size of a holopixel is typically around a square millimetre. A comprehensive description of holoprinter technology is presented in Chapter 3.

1.4 Computer Generated Holograms and diffractive optics

The vast majority of holograms are made by means of coherent interference as described in the previous section. However, this is not an absolute requirement and it is possible to calculate the diffraction patterns numerically to compute the light field based on scalar diffraction theory. CGH has been advancing at a rapid pace with the advent of increased computing power and improvements in lithographic technology to make diffractive optical elements (DOEs) and SLMs.

A review of scalar diffraction theory is presented in Appendix C which describes a theoretical framework to compute diffraction patterns under certain conditions. This sets the scene for sampling and Digital Fourier Optics. The advantage of CG approaches is that it is possible to create 2D or 3D images of objects that do not actually exist. The only limitations are the ability to describe the objects mathematically and to compute the corresponding hologram in a reasonable time. A general overview of the issues and challenges in holographic displays is presented in [28].

A diffraction pattern may be computed as a set of discrete complex values by sampling the scalar field using Fourier Optics techniques as described in Chapter 4. This yields a MxN matrix of complex elements or holopixels that may be transferred to the appropriate recording media where the media needs to record both phase and amplitude. Since CGH is a digital technique, amplitude and phase profiles are quantised into discrete levels which can be implemented as either amplitude-only or phase-only structures to simplify fabrication.

CGH techniques work best when the feature sizes of the implementation are of the order of several wavelengths for Scalar Diffraction Theory to be valid and to get a reasonable degree of diffraction. For λ=500 nm, this implies amplitude or phase features of a few microns, which is easily achievable with current micro-fabrication technology. Indeed, it is even possible to make crude holograms using modern high-resolution laser printers.

1.4.1 Amplitude-only CGH

The simplest approach is the Binary phase detour method by Brown and Lohman [63]

which records the hologram on to a transparency consisting of a matrix of cells containing different-sized apertures. Each holopixel only has two levels – either dark or transparent. The complex Fourier coefficient is represented within each cell as a transparent area whose size is dependent on the amplitude and whose position within the cell depends on the phase. These holograms are simple and quick to produce, only requiring a laser printer and a transparency film. The main disadvantage is optical efficiency as the hologram is mostly dark and high resolution printers are needed to minimise the large amount of noise resulting from the quantization of amplitude and phase.

It is possible to constrain the CGH to amplitude-only values and ignore the phase simply by taking the real or imaginary part of the complex diffraction field, applying a biasing offset, scaling and quantizing the result [34]. This yields a greyscale hologram which is difficult to fabricate in practice, so a common shortcut is to apply binary quantization. This has the effect of adding a zero-order term and a substantial amount of quantization noise as well as resulting in conjugate twin-images in the observation plane.

1.4.2 Phase-only CGH

Phase-modulation is a much more effective and efficient means of representing information than amplitude modulation. This phenomenon is experienced from everyday life where the quality of FM radio transmission is much better than AM. DOEs are transmission holograms that can modulate the complex amplitude and/or phase of some substrate material using micro-fabrication methods. Phase modulation is achieved through surface profiling and amplitude modulation is via an absorption layer to give variable transmission. The Kinoform method, proposed by Lesem and Hirsch [64], sets the amplitudes of the Fourier coefficients to unity and assumes that all of the information about the subject is contained in the phase terms varying between 0 and 2π. This is usually done with iterative Fourier Transform methods based on the Gerchberg-Saxton algorithm [65] and its derivatives, described in section 5.3.7. The phase-only assumption is surprisingly accurate, especially if the subject is sufficiently diffuse i.e. containing independent random phase terms across the samples. The kinoform leads to considerable optimisation in fabrication since it translates to a simple outline profiling of the substrate.

Silica is a popular low-cost substrate that can be etched using a series of photolithographic masks to provide a surface relief profile for phase modulation. The

phase delay experienced by light of wavelength λ travelling through a material of thickness d with refractive index n is given by:

$$\Delta\phi = \frac{2\pi(n-1)d}{\lambda} \tag{10}$$

If $\Delta\phi$ is quantised into 2^N levels and since we want to vary $\Delta\phi$ over a range of 2π, then equation (10) can be expressed as a function of the quantised step size *d*:

$$d = \frac{\lambda}{2^N(n-1)} \tag{11}$$

For example, if N=3 mask layers, λ=500 nm and n~1.5 for silica, the step size d=125nm. In this example, there are 8 steps = 1μm for a 2π phase shift. In general, a π phase shift is ~λ for silica. It has been shown that N=4 (i.e. 16 steps) is an acceptable step size for many applications and increasing the step size adds little perceptible value [66].

The simplest kinoforms are binary CGHs which simply encode two phase levels with a difference of π and can be written directly on to a photoresist [67], only requiring exposure without etching. As with binary-amplitude, binary-phase holograms also exhibit twin-image effects, however, the optical efficiency can be higher and there are no zero-order artefacts.

Other fabrication techniques for silica include reactive ion etching or ablation using excimer lasers. If amplitude information is important then an absorption profile may be created using chromium layers of a few nm deposited via e-beam lithography or direct laser writing for amplitude modulation.

A major advantage of DOEs over holograms produced by other methods described in section 1.3 is that it can produce a single on-axis image without any artefacts or intermodulation products inherent with classical interference holography. A phase-only DOE can be transferred to a metal shim which may then be used as a master in the embossing process described in section 1.6.3, ready for mass production.

1.5 Holographic Materials

The main holographic media in use today are based either on photographic emulsions or specialised polymers. Other common materials such as silica and plastics are used as substrates in certain manufacturing processes and are discussed in the next section.

Photographic film and glass plates coated with Silver Halide emulsion have always

been used in holography. The film consists of a 0.5 mm thick transparent plastic substrate whereas a glass substrate is a few mm thick. Either substrate is coated with a ~10 μm emulsion layer of photosensitive silver halide crystal grains (~40 nm) in gelatine. Both film and plates are readily available at low cost from suppliers such as Harman and Slavich. Emulsions sensitive to wavelengths across the whole visible spectrum are available. Resolutions vary between 3,000 and 5,000 lines per mm with sensitivities around 50 to 80 $\mu J/cm^2$. The emulsion is suited for both transmission and reflection holograms, although reflection holograms require higher resolution. Amplitude holograms made with this emulsion have a low diffraction efficiency of 6%, whereas phase volume holograms may have efficiencies of 90-100% [58, p. 346].

Film and plates are easy to work with under laboratory conditions, provided certain precautions regarding stability can be met. Saxby [55] provides an excellent reference on the practical fabrication of holograms. The film is typically exposed for a few seconds and is then processed chemically to develop the exposed areas and subsequently bleached to remove unexposed parts [68, p. 190]. Bleaching effectively transforms an amplitude (intensity) hologram into a phase hologram to improve diffraction efficiency [53, p. 67]. Processing can take some tens of minutes. After embossing (section 1.6.3), films and plates are the most common implementation of holograms.

Photopolymers are based on photosensitive organic materials that can be used to make volume phase holograms with high diffraction efficiencies. The material is composed of a vinyl monomer and a photoinitiation[1] compound within a film-forming polymer matrix. The monomer undergoes changes in its chemical and physical structure when it is exposed the light which transforms it into a polymer. Optically, this leads to refractive index changes in the material. Photopolymers are currently available from manufacturers such as Bayer Material Science and Dupont. The material is sensitive to wavelengths across the visible spectrum. The resolution is high at 5,000 lines per mm with a sensitivity of 30 mJ/cm^2, within a thickness of 10-30 microns, suitable for both reflection and transmission holograms. After exposure, the film needs to be developed by curing in ultraviolet light (Bayer). Some photopolymers may require baking at 100 to 200 ^{O}C for an hour or two (Dupont). There are no chemicals involved in the processing of this holographic material. High diffraction efficiencies of around 90% may be obtained. Holographic materials are widely available in monochromatic formats, although pan-chromatic materials are now becoming increasingly available for high

1 A photoinitiator is a type of photosensitive catalyst

quality photo-realistic colour holography [69].

1.6 Industrial Hologram Manufacturing

The following sections outline the most common means of fabricating holograms using the holographic materials described in the previous section.

1.6.1 Contact copying

Contact copying or printing is the simplest way to make an identical copy of a master hologram H1 on to another photosensitive material [70]. A master transmission H1 is put into contact with the copy material, usually another holographic film or plate, and the assembly is exposed to laser light. The resulting holographic copy is a true hologram with interference fringes rather than a photograph. In order to make good quality copies, the amplitudes of the primary and diffracted waves should be similar. In addition, the direction, wavelength and curvature of the original wavefront should be recreated as faithfully as possible. Contact copying is popular for low volume production with the main issues being the consistency of reproduction and quality of the copies.

1.6.2 Holographic printers

Holographic printing technologies use the principles of integral imaging where image information and the refractive lenslet are represented by a holographic pixel, sometimes referred to as 'holopixels' or hogels (for holographic elements). The basic principles of integral imaging are discussed in section 1.2.4. Holographic printers are state of the art machines that are only available from a handful of manufacturers. The devices use a variety of techniques to produce multiplexed stereographic holograms from various sources including simulated 2D computer images, from multiple angles. The holograms produced in this manner are high quality full colour reflection holograms. It is possible to produce full-parallax holograms, which lie flat, where viewer can walk 360 degrees around the image. Furthermore, holographic printers produce reflection holograms with the advantage that they can be viewed using ordinary light sources.

Holographic printers typically comprise RGB lasers, an SLM, beam steering optics and a computer to make CGH images on to the SLM described in [53, p. 242]. Each holopixel is recorded directly on to the holographic material and processed as described in section 1.5. Recent generations of holographic printers use pulsed RGB lasers to reduce the printing time from a few days to a few hours. Pulsed lasers have the added benefit of reducing the sensitivity of the printer to vibrational motion which can destroy holopixels made by continuous-wave lasers.

Holographic printers currently represent the state of the art in holographic manufacturing technology. The largest known single holographic image in 2008 is a 1.8x5.4 m full colour image of a sports car, made using a holographic printer by Zebra Imaging. The hologram comprises a 3 by 9 array of tiles, each 60x60 cm with 2.5 mm holopixels.

1.6.3 Embossing

Embossed holograms are by far the cheapest way of mass-producing holograms, using imprinting techniques similar to the production of compact discs [68, p. 229]. Indeed, the majority of holograms today are produced in this manner. Embossed holograms may be found in security, packaging and advertising applications. Common examples include holograms on credit cards, banknotes, passports or packaging for products such as batteries or consumer electronics goods.

The material comprises a thermoplastic polymer film layer, a resin separation layer and a polyester base film. A master surface relief hologram is made from metal, usually nickel, using lithographic techniques. This embossing stamp is known as a shim which is hot stamped onto a thermoplastic film layer. The film takes up the mirrored shape of the stamper and retains these features when it is cooled and removed from the press. The manufacturing process resembles stamping techniques used in traditional printing presses. For mass production, an embossing press with a heated roller can have a throughput of half a metre per second with a material width between 0.2 and 1 metre. An embossing press is depicted in Figure 19.

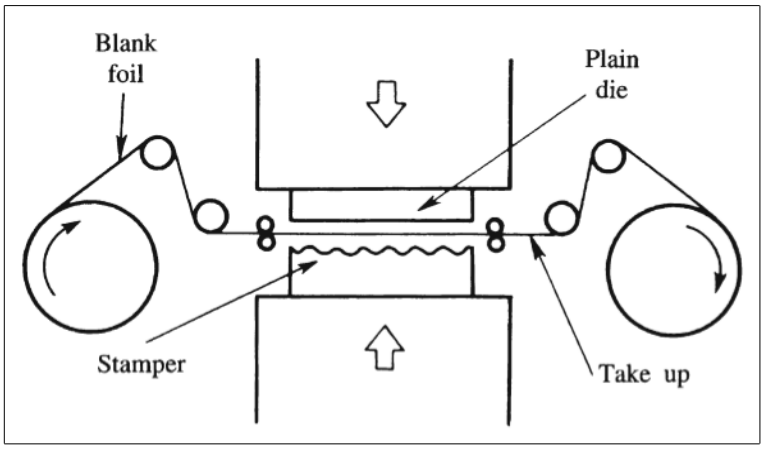

Figure 19: Embossing printing machine, from [54]

Embossed holograms are rainbow transmission holograms that can be illuminated with a white light source. Rainbow holograms are transmission holograms recorded with a slit that acts to reduce vertical parallax and thus constrain the chromatic

dispersion [53, p. 146] inherent in classical reflection holograms described in section 1.3.1. The embossed holograms include an aluminium reflecting layer deposited onto the back of the hologram film layer. This layer reflects the incident light back through the transparent hologram resulting in some kind of reflection hologram.

The embossed holograms may include an additional adhesive layer enabling transfer onto a variety of substrates that include paper or card or other polymers. Embossed holograms generally do not have the same depth as holograms produced by other methods. The depth is constrained to a few centimetres. The images tend to be a combination of 2D and 3D images or 3D images composed of 2D components using CGH and integral imaging techniques.

1.7 Technical approach taken in this thesis

Holographic display research has largely been driven by science fiction, ever since the first Star Wars movie. This has proven to be a very difficult technical challenge for researchers when they are faced with practical engineering realities. The conventional approach to 3D displays by the scientific community (section 1.2.5) is a top-down view; the work advocated here takes the opposite, bottom-up, approach. Hence, the starting point for the research proposed in this thesis is the question: what is the simplest holographic display that can be made? The answer is simply to make a single-voxel display and switch the voxel on or off. Such a hypothetical display is trivial and not particularly interesting, so the next step is to scale this to a two-voxel display and switching either voxel on or off. This is more interesting than the single-voxel case but it is still somewhat trivial and could be accomplished in several ways. For example, it is possible to simply selectively illuminate holograms that are arranged in a side-by-side fashion. This has the disadvantage of a loss of field view and scalability is limited as more lights are required for reconstruction. A more elegant approach could be to use angular multiplexing, via multiple reference beams [53, p. 162], and to selectively illuminate via angular reconstruction. This would solve the field-of-view problem but the need for multiple light sources is still a limitation. Furthermore, the holographic material requires multiple exposures which leads to cross-talk and saturation, thus limiting the scalability of this method. The thought-experiment can be extended further to tens, hundreds or more voxels. This is where conventional techniques break down and other approaches are needed.

This project is about the research, design, development and testing of three dimensional volumetric holographic displays based on Holoxica's proprietary

holographic screen technology. The approach promises relatively low cost 3D volumetric holographic displays through leveraging existing and state of the art holographic manufacturing techniques in novel ways. These techniques span classical holography, holoprinting, holographic optical elements and CGH/diffractive optics.

The 3D displays work on the volumetric principle where a series of voxels are placed adjacently in 3D space, each of which may be addressed independently. This means that each holographic voxel may be switched on or off (i.e. visible or not) and its colour may be changed. In this thesis, "voxel" is defined somewhat broadly to encompass image segments, icons, symbols, characters or dots. The technology is capable of producing full colour moving images through the temporal manipulation of any number of such voxels. The technology can also be used to enhance current high resolution static holograms with display-like features.

The core technical approach is based on the observation that complex objects can be made through rendering collections of simpler volume elements or voxels rather than attempting to render whole objects or slices of objects. The concept uses a new way of recording traditional holograms such that a series of holographic images can be spatially multiplexed orthogonally to make a holographic screen. The process essentially samples holograms and then puts several different sampled holograms together to make a singe composite hologram. This method leads to a substantial amount of simplification in terms of hardware, bandwidth and computation compared with other state of the art approaches to holographic displays described in section 1.2.5.

1.7.1 Project objectives

The objectives of the project are to:

- Explore the limits of the hologram sampling and interleaving approach
- Decrease the cost through leveraging alternative implementations
- Maximise visibility in terms of brightness and viewing angles
- Realise colour
- (Trans) portable displays
- Realise 'human-sized' images
- Develop a theoretical model and simulation environment
- Develop designer-friendly tools and methods
- Maximise resolution
- Dissemination to the scientific community and the public
- Intellectual property protection

These objectives are ambitious and challenging, covering a wide range of activities. The objectives essentially explore the limits of the technology as well as commercial factors such as cost, quality and manufacturing capability.

1.8 Outline of the thesis work

The technology to be researched combine volumetric and holographic approaches described in sections 1.2.3 and 1.2.5. The project aims at making commercially feasible holographic volumetric 3D displays with a fixed number of frames. However, the trade-off for this approach is the limited number of frames within the 3D display. Nevertheless, these displays may still be used in a large range of applications.

Chapter 2 describes the basic principles behind the fixed-frame holographic approach and applies the sampling and interleaving method to classical analogue holographic techniques. This is used to realise a first generation display based on transmission holograms and laser illumination. The holographic display is capable of showing simple numerics and segmented digits in various configurations. The design of the holographic screen is relatively straightforward but the implementation of the holographic screen is more difficult as it is done manually. The holographic display is realised with a laser projector.

Chapter 3 extends the sampling and interleaving concept to digital holography using holographic printing technology. The design of the hologram is more complex than the previous display, using 3D computer graphics tools and holographic screen implementation is performed automatically via a holoprinter. The holoprinter needs to be reconfigured to perform the sampling and interleaving purely digitally in software. The holoprinter produces a holographic screen as a digital reflection hologram. The holographic display uses an amplitude masking approach based on a transmissive liquid crystal display with white-light illumination.

Chapter 4 takes the digital concept a stage further into the realms of diffraction theory and computational Fourier optics. The design of the hologram is performed with a series of custom algorithms to realise purely digital holograms that can be reproduced on an SLM. A further set of tools can take the design and produce masks for photolithographic production of DOEs. A simple DOE is fabricated on a glass wafer and characterised.

Chapter 5 takes the algorithms from Chapter 4 and explores them in more detail in terms of sampling popular signals and propagation modelling in Matlab. Chapter 6 channels all of the algorithms into a comprehensive framework aimed at the exploration and design of holographic voxels. This leads to FOURIOT, a generalised Fourier Optics

modelling tool that can be used to generate diffraction patterns for arbitrary input shapes, propagate these to the hologram plane and do a reverse-propagation to an observation plane. The user can choose amongst a large array of options including image properties (type, size, resolution), illumination wavelength, position in z-space, quantisation (phase/amplitude), propagation models, optimisation models, metrics and output to various formats (including SLMs for verification).

Chapter 7 applies the sampling and interleaving concept to the implementation of a purely digital computer-generated holographic screen. It builds on the algorithms from Chapter 6 to make a simulation environment around the holographic screen, and the imagery from Chapter 2 is repeated and expanded to realise a dual seven-segment style display purely in a simulation. The holographic screen is then implemented on an SLM for physical verification. The process shows the validity of the spatial multiplexing approach from the ground up including theory, simulation and implementation.

Chapter 8 shifts the approach from Fourier Optics to Fourier Imaging and proposes a novel holographic display architecture based on a large-aperture holographic optical element. This enables arbitrary high-resolution images to be displayed at discrete depth planes, depending on wavelength, in real space. The display system is coupled to a motion sensor and it is possible for the viewer to interact with the display in mid-air almost in real time.

Chapter 9 concludes with a summary of the alternative approaches to realising holographic displays opened up by this work. This leads to numerous avenues of research which bring the prospect of a true holographic display away from science-fiction fantasy and into the realm of engineering reality.

Chapter 2 - Classical Interference Holography

This chapter describes the realisation of the first generation proof-of-concept holographic volumetric display. The 3D display works on the volumetric principle where a series of voxels are placed adjacently in 3D space and each voxel may be addressed individually. This means that each holographic voxel may be switched on or off (i.e. visible or not) and its colour changed. Voxels may consist of image segments, icons, symbols or characters. The core technical approach is based on the observation that complex objects can be made through rendering collections of simpler volume elements or voxels rather than attempting to render whole objects or slices of objects.

The means of fabrication is based on adaptations of classical analogue holographic techniques. Firstly, the principles of hologram sampling are investigated through masking classically recorded holograms. The concept uses a new way of recording traditional holograms such that a series of holographic images can be spatially multiplexed orthogonally to make a holographic screen. The process essentially samples holograms and then puts several different sampled holograms together to make a single composite hologram. This method leads to a substantial amount of simplification in terms of hardware, bandwidth and computation compared with other state of the art approaches to holographic displays

2.1 Sampled hologram approach

A hologram contains an interference pattern of the 3D image across its entire surface area. If one part of such a hologram is removed then it still contains the 3D image but only from a unique perspective with limited resolution and depth of field. Another way to do this is to illuminate a single portion of the hologram, which reconstructs a unique view of the stored 3D image. If the illumination spot is moved quickly in a raster-scanning pattern across the entire surface of the holographic plate then the entire 3D image would be reconstructed, albeit somewhat fuzzier and more blurred than the original image. This thought-experiment shows that a hologram can be sampled through temporal scanning. An alternative means of sampling a hologram is spatially where the single scanning hole in the previous example is replaced by a grid of static holes or a simple mesh. It can be shown that if this mesh is fine enough then the original image can be reconstructed. Further, if several distinct holograms are sampled by different grids then it is possible to interleave or spatially multiplex a number of holograms together. This principle is illustrated in the next section and a rigorous mathematical treatment of sampling is presented in Chapter 5.

2.2 Principle of Hologram Sampling and Interleaving

The principle of spatial sampling and interleaving to make a holographic screen is illustrated in Figure 20. This depicts two different holograms (interference patterns) of a cube and sphere shape sampled using a pair of masks. The checkerboard mask patterns correspond to complementary sampling grids i.e. mask 1 is the inverse of mask 2. This means that the resulting sampled holograms are spatially orthogonal. Consequently, the pair of sampled holograms can be interleaved to form a single composite hologram. It can be seen that the composite hologram contains the interference patterns corresponding to the original ones. This fabrication method forms the basis of the holographic screen.

A simple two-element holographic screen is described here where the cube and sphere correspond to simple volume elements. The concept can be extended to record more volume elements on a holographic screen using an extended set of sampling grids and

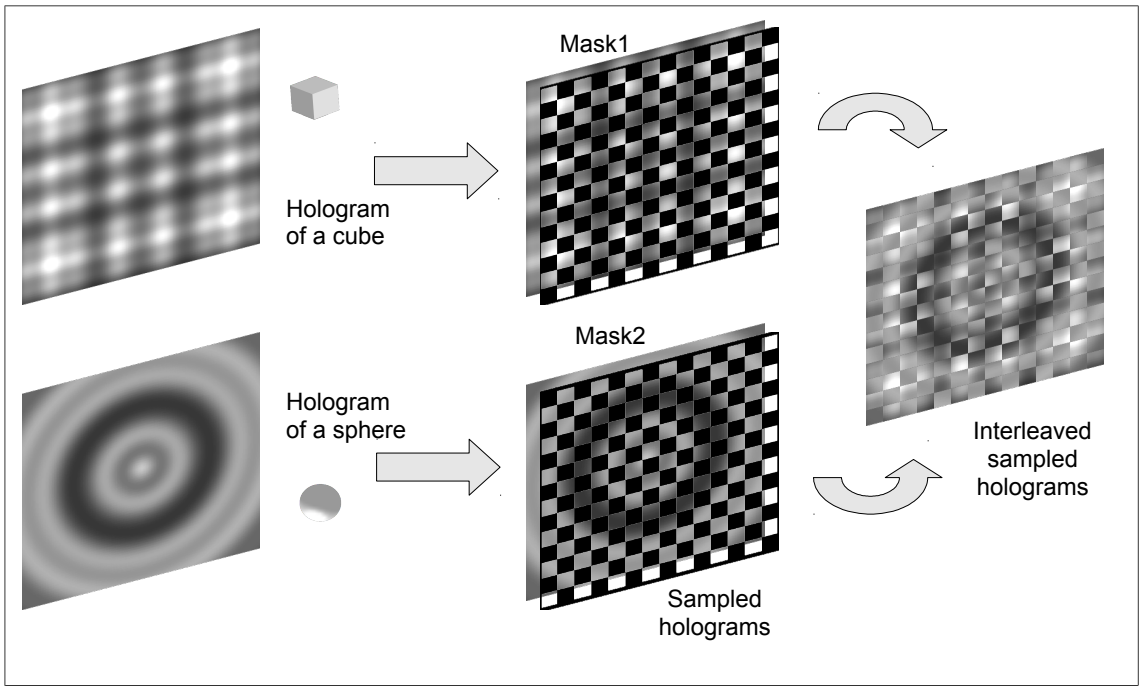

Figure 20: Sampling and interleaving holograms

masks. It is evident that the positioning of the volume elements can be flexible in three dimensions within a prescribed zone both outside the plane of the holographic screen, as in the case of real images, or behind the screen for virtual images. Each square within the sampling grid can be considered as a holographic pixel or holopixel which reconstructs a portion of the corresponding 3D image from its own unique viewpoint. A 2D array or matrix of such holopixels can reconstruct the whole image from a range of viewpoints.

The process presented here can be applied to a variety of hologram fabrication methods and holographic media including silver halide film and photopolymer (section 1.5). Some variations on the standard hologram recording process are required to accommodate the sampling and interleaving mechanism. The process is applicable to reflection as well as transmission holograms. The focus of this chapter is on transmission holograms because these are easier and less costly to fabricate than reflection holograms. Reflection holograms are considered in the next chapter.

2.2.1 Holographic screen fabrication

This section describes the steps involved in making four and nine image holographic screens. The four-image holographic screen presents numeric images from 1 to 4. These are recorded in three configurations including side-by-side, spatially overlapped and in-line (along the z-axis). The nine-image holographic screen is a segmented display bearing the popular seven-segment arrangement together with two additional segments.

The holographic screens are fabricated using a variant on classical holographic recording methods described in Saxby [55, p. 46] (also section 1.3.1) which have been modified to include extra processing stages. The off-axis transmission hologram recording technique is based on a silver halide film which is selectively masked and exposed to record each voxel segment (numeric images or segment).

The recording setup is illustrated in Figure 21 comprising the silver halide hologram plate H with a mask M, which records the object interference pattern. During recording, the mask is in contact with the holographic plate on the emulsion side. The object image consists of numerics or segments which can be either replaced or taped off (with black masking tape) in between successive exposures. The position of the object image can be changed in between recordings to place the voxel in different spatial locations. The object image is placed behind a ground-glass diffusion screen to scatter the object beam, reducing speckle on reconstruction. Each recording sequence comprises the following steps:

1. Place mask M1 in front of hologram plate H
2. Place object transparency T1 in front of diffuser at position $(x_1,y_1,z_1)$from H
3. Expose H by switching on the reference and object beams

The recording sequence is repeated by replacing the mask to M2 in step 1 and changing the object transparency T2 (and optionally, its position) in step 2. There are

four recording sequences for a four-image holographic screen and nine sequences for the nine-segment screen.

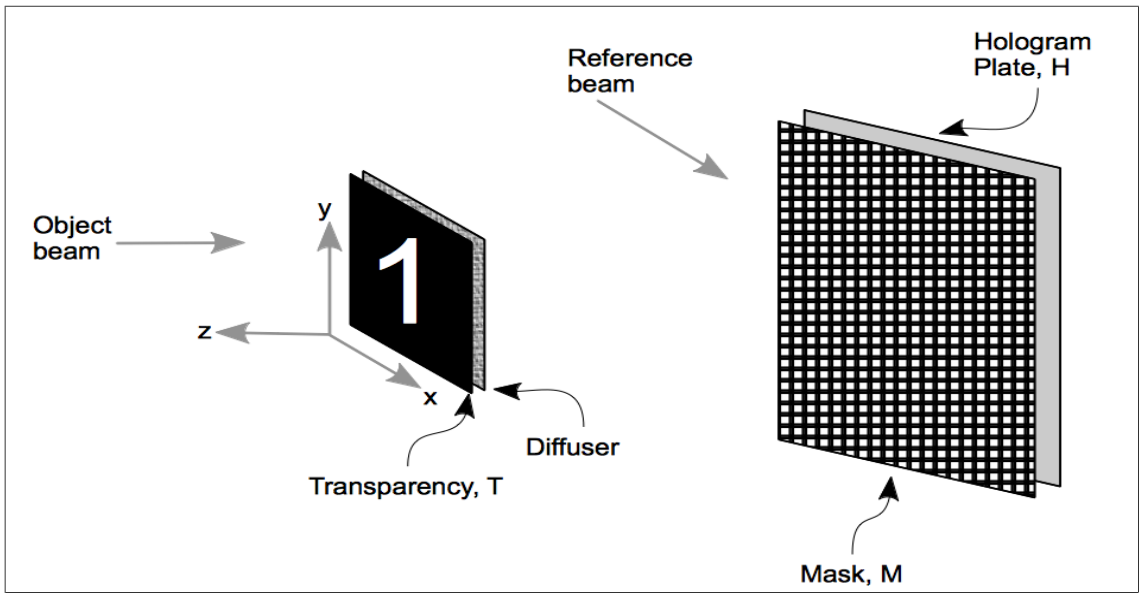

Figure 21: Holoscreen recording configuration

The segmented digits for the 9-segment holographic screen are depicted in Figure 22 where the rightmost digit comprises the classic seven segments labelled a to g. Two additional segments labelled h and i are shown on the left and these are composed of whole numbers i.e. the actual digits '1' and '2' similar to the digits described earlier. In fact, the segment combinations 'h' and 'i' are spatially overlapped. This arrangement brings the total segments up to nine, conveniently representing a 3x3 sampling grid. Such a display would be capable of counting up to 29.

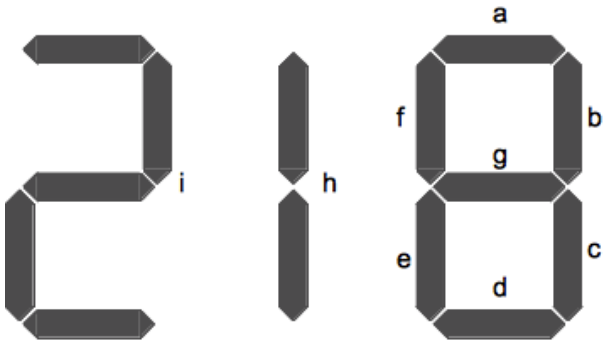

Figure 22: Design of segmented digits

The masks for both numerics and segmented holoscreens were designed using the free OpenOffice Draw graphics package. These were printed on to a transparency via a 600 dpi laser printer before being transferred lithographically on to glass plates. The object

transparencies for the numeric and segmented screens are presented in Appendix A. The following holoscreens are fabricated:

- Numeric screen with four digits spatially overlapped
- Numeric screen with four digits side-by-side
- Segmented screen with nine segments

The holographic screens were fabricated by Iñaki Beguiristain of Display Hologram, a UK-based specialist supplier of custom holograms. The Agfa silver halide film is chemically developed and bleached before being dried and laminated onto a glass plate. The resolution of the holographic film is of the order of 3,000 lines per mm and all recordings were done with a 10 mW HeNe laser at 632.8 nm.

The holographic screens are about the size of an A4 page (20x30 cm). The holopixel size is 4x4 mm which means a sampling grid of 50x75 holopixels. These dimensions were chosen empirically for ease of fabrication to simplify the alignment of masks between exposures. The screens were fabricated in transmission mode for simplicity and reliability.

2.3 Holographic display system

Traditional holography has always relied on a single point light source for

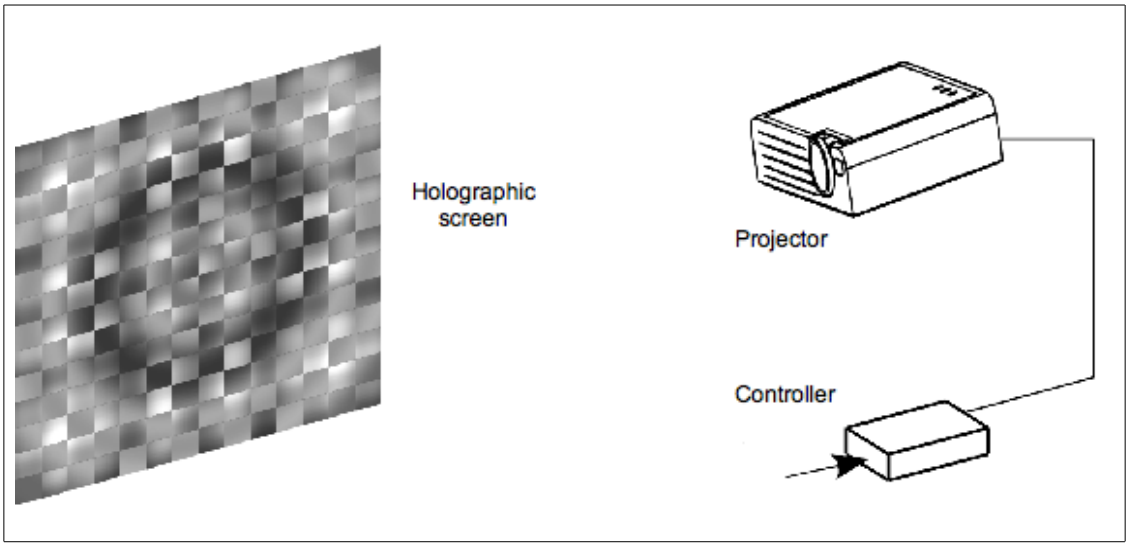

Figure 23: Holographic display configuration for playback

illumination which is either a spotlight or LED in the case of a reflection hologram or a laser for transmissions. Here, this light source is replaced with a projector which shines patterns on to the holographic screen to reconstruct any or all of the pre-recorded

images. The projected patterns correspond to combinations of the grid masks used to fabricate the holographic screen. The playback hardware configuration is depicted in Figure 23.

The control unit drives the projector to produce a series of patterns on the holographic screen, as shown in Figure 24. These patterns may be used to select the areas on the screen that contribute to the reconstruction. The first pattern shows reconstruction of the cube shape by projecting the first mask, followed by the sphere and the second mask. If the whole screen is illuminated then both stored images are reconstructed.

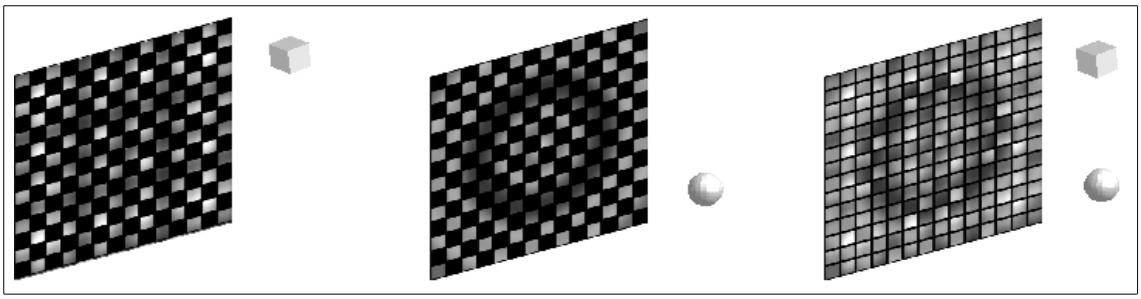

Figure 24: Reconstructing stored images via the holographic screen

2.3.1 Hardware configuration

The laser projection device used for the experimentation and measurements is the first commercial laser pico-projector evaluation kit made by Microvision. The PicoP evaluation kit is a three-colour laser device (640, 532 and 440nm), WXGA resolution (848x480). The device is based on a MEMs micro-mirror and has a diameter of about 1 mm with biaxial deflections of 43.2 by 24.3 degrees. The projector has an infinite focal length with no optics in the beam path. The spot size is roughly linear with distance, which is 1 mm at 1 m, or 0.5 mm at 0.5 m [71].

The device is driven via an ordinary analogue VGA port from a computer. At first, an ordinary netbook PC was used to drive the projector but this was soon changed to an embedded Single Board Computer (SBC). The SBC is the Kontron JRexplus based on a x86-compatible AMD LX800 fanless CPU at 500MHz, 1GB of RAM and a variety of standard interfaces: VGA, USB, Ethernet etc. A 4GB compact flash card is used as a bootable hard disc to hold the operating system and software. The board consumes less than one watt of power.

2.3.2 Software and programming environment

The digital hardware (netbook or SBC) run the open-source Linux operating system, based on the Debian and Ubuntu distributions. The code is written in Python version 2.7, an open-source programming language that is scripted so that it can be run

immediately without compilation. Python is an open source language with plenty of documentation and support libraries across many computing platforms [72,73]. This means that code development can be done on a workstation and it will run on any computer, including the embedded platform, without any further modifications.

The versatile PyGame graphics and gaming library version 1.9.1 [74] is used to access and control the video interface at the bit level, including full screen control plus support for direct keyboard control and timing functions. This library is also available across multiple computing platforms. Although not strictly necessary, a lightweight window manager, XFCE4, was installed, in order to ease installation and development.

The development environment used for coding is the commercial Wingware Integrated Development Environment or IDE (version 3.x). This is one of the best Python IDEs around with an integrated debugger, version control and project management tools. Development in Python has the advantage of fast turnaround for research and prototyping purposes. The language itself is quite compact and elegant which leads to succinct and fewer lines of code compared to others such as Java or C/C++. However, the penalty for all of these desirable features is reduced performance compared to compiled languages such as C/C++. Nevertheless, most current computers have efficient Python interpreters and plenty of CPU and memory resources to run the code without severe performance degradation.

2.3.3 Code for mask pattern generation

A functional description of the code and algorithms is presented here, from a user's perspective. A version of the actual Python code is presented in Appendix B. The code runs directly from the command line with the following parameters:

```
> python testpatterns.py default.cfg patterns.seq [-f]
```

The command invokes the Python interpreter to run the program testpatterns.py which normally requires two text files: a default configuration file and a pattern sequence file. If no parameters are specified then the program chooses a default set. The optional [-f] flag starts the program in full-screen mode instead of windowed mode. Normal operation should be full-screen mode whereas windowed mode is for testing.

On startup, the program shows the following menu in the terminal console:

```
Arrow keys: offset (x and y)
< > keys: margin spacings
- + keys: size of tile
[spacebar]: next sequence
g: grid on/off
k: keystone on/off, a & z: keystone up/down
d: dump configuration parameters to a file
s: sequence ON/OFF (timed)
ESC/q: quit
```

Most of these commands are used for calibration and diagnostic purposes where the user can generate, move and scale the mask patterns displayed on the laser projector. The idea is to align the patterns generated by the projector with the features on the holographic screen.

A typical default.cfg file contains the following parameters:

```
W=800
H=480
M=10
margin=1
offsetX=7
offsetY=7
scaleX=1
scaleY=1
```

The [W,H] parameters describe the display resolution (848x480 for the WXGA laser projector), M is the size of a sub-tile in pixels, margin specifies the spacing between tiles in pixels, offsetX&Y is an offset for the entire screen from the origin in pixels and scaleX&Y define the pixel aspect ratio. All of these parameters can be adjusted manually using the keyboard controls e.g. the four arrow keys can adjust the offsets in four directions (up/down/left/right), +/- keys change tile sizes etc. Once the screen has been correctly adjusted, the parameters can be dumped to a configuration file via the 'd' key. This file is typically named *default-n.cfg* where n is a sequence number.

The default configuration assumes a 2x2 tile geometry. The sequence text file for a 4-segment display with a 2x2 tile geometry typically looks like:

```
[tiles]
geometry=2,2
pattern1 =1,0,0,0
pattern2 =0,1,0,0
pattern3 =0,0,0,1
pattern4 =0,0,1,0
pattern12=1,1,0,0
pattern123=1,1,0,1
pattern1234=1,1,1,1
```

```
[sequence]
pattern1=white
pattern2=blue
pattern3=red
pattern5=green
```

The first [tiles] section defines the tile geometry as a 4-element 2x2 matrix $\begin{bmatrix} a & b \\ c & d \end{bmatrix}$ and each individual tile pattern name (pattern1, pattern 2 etc) is followed by a sequence of 4 bits {a,b,c,d} that map on to this matrix. For example, the first sequence {1,0,0,0} maps to a tile thus: pattern1 $= \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}$ which means the top left corner is white and the rest is black. The parameters from the previous configuration file are applied to this tile (pixel size, offsets, margins etc) in order to generate the whole screen bitmap. From the above, pattern1234 has all segments illuminated, meaning that the whole screen is white (pass through).

The second [sequence] section contains pattern and colour pairs in which each pattern is displayed as the corresponding colour in sequence when the spacebar is pressed. The code can automatically cycle through all sequences with a delay of one second per pattern via the 's' keyboard option. Once the sequence reaches the last pattern (pattern5), it loops back round to the top (pattern1).

2.4 Results and observations

The fabricated holographic screen is about the size of an A4 page, depicted in Figure 25. The pixellated structure of the mask patterns can be clearly seen.

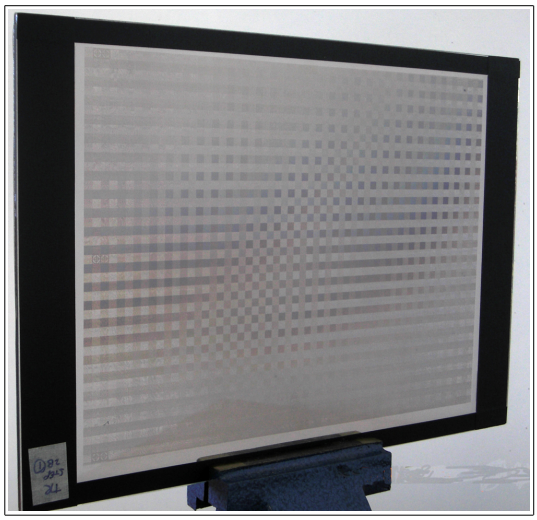

Figure 25: Holographic screen

The holographic screen is illuminated by the Microvision pico-projector that illuminates an array of holographic pixels across the plane of the screen. The optical setup is depicted in Figure 26 where a pattern of dots from the laser projector can be seen on surface of the holographic screen. The figure does not show the digital hardware controller.

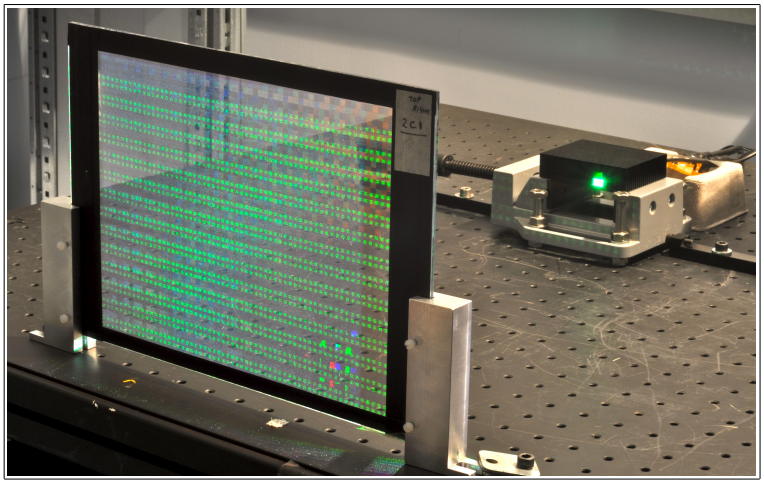

Figure 26: Optical setup for replay with laser projector

2.4.1 Numeric holographic screens

A holographic screens depicting the numbers '1' to '4' was fabricated in spatially

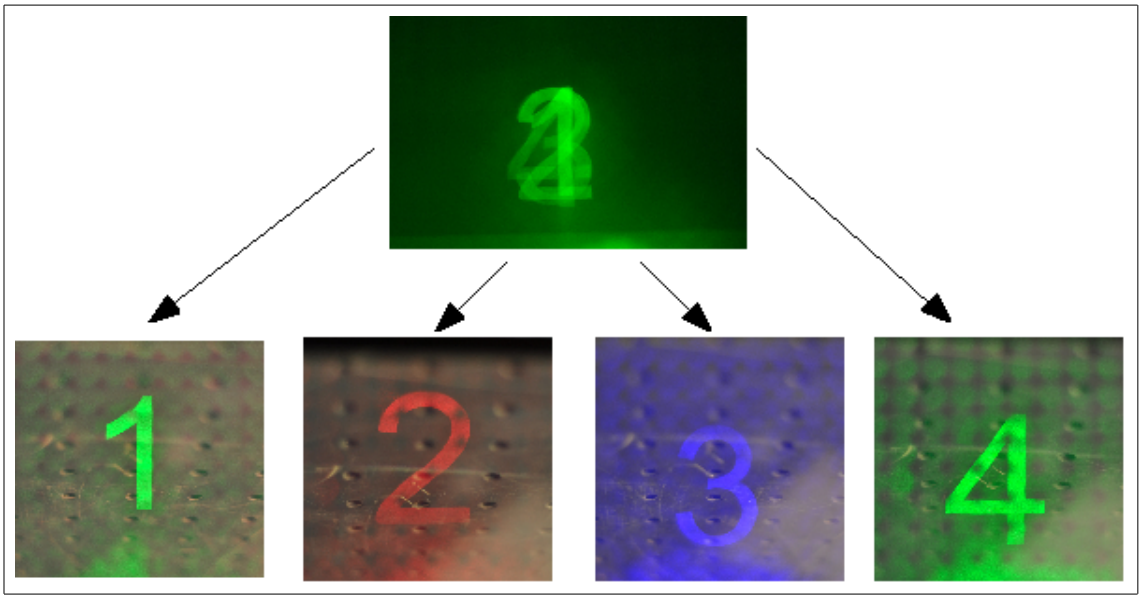

Figure 27: Spatially overlapping holographic digits in three colours

overlapping configurations Figure 27. The sampling grid topology is a 2x2 matrix i.e. four holograms are multiplexed on these screens. Any combination of numbers can be

reconstructed by changing the pattern of laser light on the back of the holographic screen; and the colour can be switched between red, green or blue.

The holographic screen of Figure 27 appears rather confusing when fully illuminated since all of the volume segments are overlapping, as shown in the topmost image. The individual digits can be selected by illuminating the holographic screen with the appropriate mask patterns via the laser projector. Any combination of digits can be selected in any of the three colours. The display can count up and down in real-time. The figure only shows a subset of all the possibilities.

The next holographic screen depicts the same set of numbers in a side-by-side configuration, Figure 28. All other parameters such as the sampling grid topology and patterns are the same as the previous overlapping screen. Once again, it can be seen that any singular digit can be selected or even combinations of digits because there are no confusing overlaps in this case.

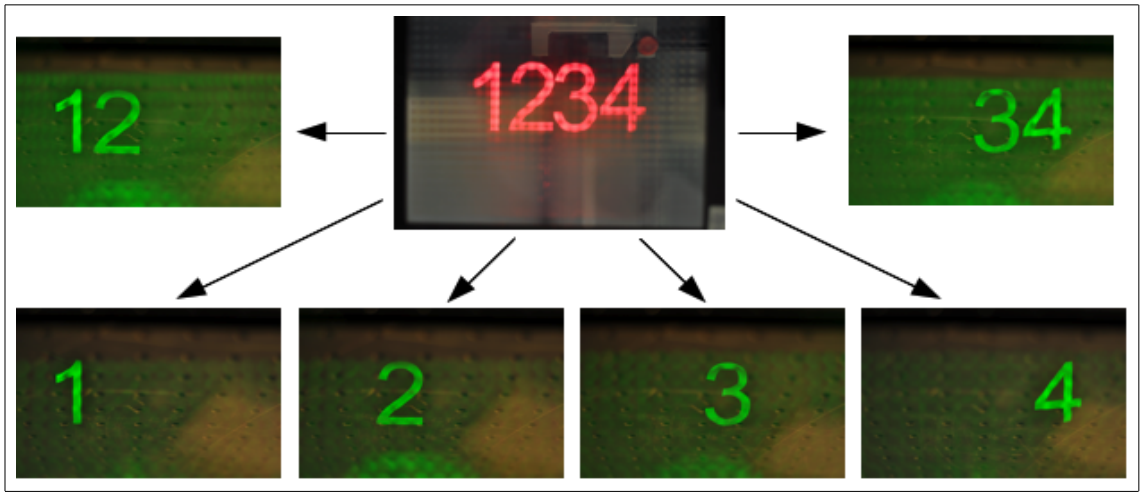

Figure 28: Adjacently placed holographic digits

The display performance characteristics in terms of brightness and contrast ratio arrangement were measured with a Minolta CS-100 Chroma meter. The two-image holoscreen (not shown here) and the four-image adjacent digits (Figure 28) were measured. Since it is not possible to show pure white voxels with such a display, because it is a chromatically dispersive transmission hologram, the measurements were performed for green only. The luminance meter did not have sufficient sensitivity to measure pure black in the laboratory, so a small amount of ambient light was introduced to get a reading for the contrast measurement. The laser projector was programmed to illuminate all segments for the brightness reading and all-black for the contrast ratio.

The average ambient light level was measured at 0.018 cd/m^2. The average brightness for the 2-digit display was 56.7 cd/m^2 with a contrast ratio of 3092. The average

brightness of the 4-digit display was 9.2 cd/m^2 with a contrast ratio of 500.

2.4.2 Segmented holographic screen

A holographic screen was fabricated as a segmented display in the style of the popular seven-segment format. The sampling grid topology was a 3x3 matrix with nine holographic segments. The total height of a segmented digit representing the number '8' is 11x6 cm, which is recorded 20 cm outside the plane of the holographic screen. The upper image in Figure 29 illustrates the segmented screen with all segments illuminated

Figure 29: Segmented holographic digits

under ambient lighting conditions.

This display can be programmed to produce the classic seven-segment style of patterns through illumination of certain combinations of segments. The lower images in Figure 29 depict a series of segmented digits in various colours, photographed in virtual mode and inverted. Again, the concept can be taken further with temporal manipulation to yield a counting display together with colour manipulation. The display is capable of counting up to twenty nine.

2.5 Acknowledgements

I would like to acknowledge Iñaki Beguiristain of Display Hologram for his patience and skill in fabricating the holographic screens. I also acknowledge the contribution of Mikko Halonen of Holoxica for setting up the segmented holographic screens, performing the alignment, specification of tile geometries and pattern sequences to perform the counting functions; all based on my code. He also took the photos in Figures 27 to 29.

2.6 Conclusions and further work

If a hologram is broken down into a multitude of pieces, each piece will contain

information about the scene from its own unique viewpoint. This implies that holograms have a great deal of redundant information that can be exploited through sampling. The work has shown that multiple holographic images can be encoded through spatial multiplexing and interleaving with complementary mask patterns to record a composite holographic screen. The holographic screen is then used as a component in a display system with a laser projection device that simultaneously provides illumination and playback patterns coincident with the recording mask patterns.

The hardware setup is simple and low-cost, comprising off-the-shelf commodity components. The reconstructed images are large and highly visible under ambient laboratory lighting conditions. Good results were obtained for a relatively coarse-grained sampling grid on the holographic screen where up to nine holographic images are sampled and interleaved. There is no speckle observed in the reconstructed images, which is another good result; due to the diffuser used in the recording process, together with the de-speckling mechanisms used in the laser projector. The brightness measurements show that the display is visible under ambient laboratory conditions, although a dim environment is better. The contrast ratios are good and are likely better than the measured values due to the sensitivity of the instrument.

The reconstructed images were photographed and videoed with a Nikon D90 digital SLR camera, which has good performance under low light conditions and it can also take high definition movies. Of course, it is difficult to show the true 3D effects of the holographic display with photographs or video so the visual impact is diminished. Reconstruction is done in real mode where the images are floating in mid-air, as well as virtual mode where the images are behind the screen. In the results presented here, virtual images were more manageable for taking photographs and videos in terms of a wider viewing angle. The optical setup in Figure 26 is for real images whereas re-using the same setup for virtual images renders the reconstructed images upside-down. Hence, the corresponding photos have been inverted for convenience. Apart from some cropping, no other image processing or enhancements were performed on the photographs.

The initial version of the Python software comprises some 460 lines of code, excluding comments. The code has been written for functionality and it does not include extensive error checking. Although the code is not known to crash at runtime, it will exit the Python interpreter if there is an error. Future versions shall include more robust error-checking. Although it is said that Python is unsuitable for realtime

performance, this has not been observed here and it is perfectly adequate for this application.

This work was presented at the SPIE Photonics Europe 2010 conference, held in Brussels, and published in the proceedings [75]. A patent on the display system, masking scheme, methods of recording the holographic screens and reconstruction schemes was submitted to the UK and the European Patent Office [76].

2.6.1 Future work

The current holopixel dimensions are rather large (4x4mm) and these could be reduced dramatically up to two orders of magnitude, or even further, ultimately down to the diffraction limit of the laser projector spot. This point is tackled in subsequent chapters. The work shows that the laser projector has sufficient temporal coherence to perform holographic reconstruction simply by sweeping the laser spot across a hologram. Given that these devices are currently mass-produced at low cost, this could be an interesting and potentially compact means to illuminate transmission holograms without resorting to beam-expansion and collimation optics. I have not come across such an illumination strategy in the literature. This notion could be taken a stage further by including the laser projector within the adapted recording process where the laser source could be split with one arm feeding the projector for the reference beam and the other used as the object illumination beam. The projector could produce the mask patterns directly, thus eliminating lithographic plates from the recording process. This would pave the way for a digitally programmable and highly flexible holographic screen recording apparatus.

The work shows that the laser projection means of illumination is not ideal from a stability and reliability viewpoint. The laser projector beam is subject to thermal drift as the MEMs micro-mirror is warmed up by the incident laser beams. It is relatively easy to push either the laser projector or the holographic screen out of alignment with relatively small knocks or movements. The mechanical configuration therefore requires a rigid setup and the setup requires a calibration step almost every time it is used. These issues are addressed in the next chapter where an alternative optical illumination architecture is explored, based on a laser-illuminated liquid crystal optical shutter mask.

The real-space image produced by the transmission hologram is currently pseudoscopic since it is recorded in a single step to save on costs. This means that the reconstructed real-space image behaves strangely as the viewer moves around it, although it appears fine in a video. This is sufficient for proof-of-concept purposes but it

is unsuitable for real-world applications that require an orthoscopic real-space image. There are two ways to achieve this, the classical method being the two-step H1-H2 method where a hologram of a hologram is made where two pseudoscopic recordings yield an orthoscopic image. Although popular, this method is rather long-winded, expensive and cumbersome, requiring a great deal of work to make two sets of hologram recordings. A more elegant single-step method is to use a converging lens to image the object and to make a transmission hologram of the image in the normal manner. The main drawback is the size of the lens which would need to be of similar dimensions to the holographic plate (~20 cm diameter) with a short focal length or small f# in order to image the digits/segments presented here. Although large lenses do exist and are used in specialist applications e.g. astronomy, they are somewhat expensive; however, a more affordable alternative is to use parabolic mirrors or Fresnel lenses [53, p. 125]. Fresnel lenses are often made of acrylic, are readily available and found in overhead projector optics and simple lighting applications. This is a practical and viable option that could be explored in the future.

The work shows that there is an alternative means to make simple holographic displays that do not require heavy computational resources, complex electronic or optical setups. These displays are straightforward to make using modified holographic fabrication methods and commercial off-the-shelf components. Also, there is plenty of scope for further refinement and improvements.

Chapter 3 - Holoprinters and Digital Holograms

The previous chapter focussed on the development of holographic screens based on transmission holograms using classical manual holographic techniques. Although the results were good, the holographic screens are difficult to fabricate or replicate and the process very labour intensive, which makes them quite expensive at a few thousand pounds per screen. Scalability in terms of multiplexing more images into the holographic screen is also limited since the process is risky and error-prone. Furthermore, the screens are based on transmission holograms that require laser illumination. This again is not scalable to larger-area displays, necessitating increased laser power, leading to laser safety concerns. Furthermore, colour is not easy to realise with transmission holographic screens and this would entail separate red, green and blue holopixels, thus increasing the complexity by a factor of three. Some of these issues are revisited and tackled in the next chapter.

The principle of hologram sampling and interleaving discussed in the previous chapter is applicable to both transmission and reflection holograms. The work in this chapter addresses the shortcomings of the previous methods to make holographic screens based on reflection holograms which only require white light illumination. This eliminates the problems associated with laser illumination, although it introduces new issues. In order to overcome the shortcomings of manual production, the holographic screens are fabricated automatically using a holographic printing machine, or holoprinter. Holoprinters produce full-colour digital holograms through encoding red, green and blue interference patterns that overlap within the same holopixel. All of these advantages make the holoprinter approach very compelling.

Whereas the previous chapter presented an empirical approach to the design and fabrication of the holographic screens, this work will take a more formalised approach. The work will first create a mathematical framework for the generation of holographic screens via sampled and interleaved digital holograms for fabrication by the holoprinter. The digital holograms are themselves composed of a set of holographic pixels or holopixels, so digital holograms can be considered to be a sampled representation of the 3D scene or image. In this work, an extra layer of sampling is introduced on top of this already-sampled image.

3.1 Overview of holoprinter technology

King in 1970 showed that a digital hologram could be synthesised purely from a series of two-dimensional views [77] derived either from a rendered computer model or a

series of photographs. This was an important step as it meant that the hologram could be separated from a physical object or scene; so these could be purely virtual or synthetic. The first holoprinter devices made in the 1970 and 1980s were two-step printers [78] where a master composite hologram is written, known as a H1 hologram that is subsequently copied holographically, referred to as a H2 transfer hologram [68, p. 40]. One of the first groups to implement a single-step direct-write digital hologram was Yamagushi [79] in the early 1990s. Later improvements included the introduction of colour and faster printing with pulsed lasers [80,81].

Holoprinters are state of the art machines, costing up to around half a million dollars. There are only a few digital holoprinter manufacturers in the world, including Geola (Lithuania) and Zebra Imaging (US). Other companies active in the field include View Holographics (UK) and a startup Ceres Imaging (UK). An example of a holoprinter machine is depicted in Figure 30.

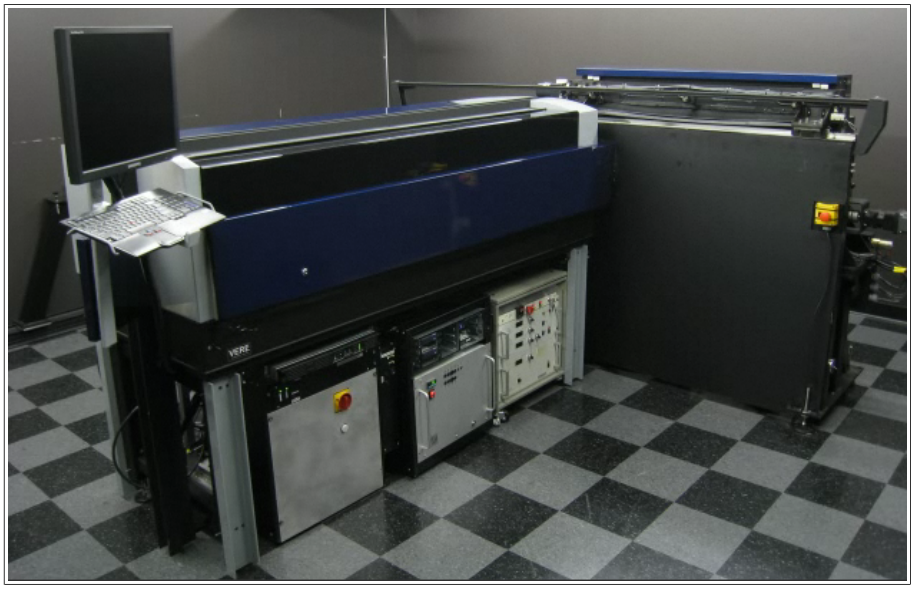

Figure 30: Holoprinter device, courtesy of Geola

3.1.1 Optical architecture of a holoprinter device

The digital hologram comprises a matrix of holographic pixels, or holopixels, created via the interference of red, green and blue lasers in a standard holographic medium such as photopolymer or silver halide on film (see section 1.5). The digital hologram is sometimes referred to as a stereogram. A simplified schematic of the optical architecture of a typical holoprinter [82] is depicted in Figure 31. The laser writing scheme comprises an object and reference beam pair as with classical holography (Chapter 2,

section 1.3.2). The object beam is modulated via an LCD micro-display which presents information synthesised from the scene. The optics are held stationary whereas the hologram medium is mounted on an X-Y translation stage and moved relative to the lasers. The lasers are usually arranged vertically in-line and they operate on different holopixels in parallel. The stage is moved continuously in a raster-scan fashion horizontally (x-direction) until the end-stop where the hologram plane is moved up a step (y-direction) and the scan is repeated in the opposite direction. The setup ensures that the same holopixel gets written by the correct red, green and blue images. The digital hologram is composed directly in a single step. The lasers, micro-display images and motion of the X-Y stage/holographic medium are all synchronised via a computer controller (not show in Figure 31). Printing a digital hologram in this manner can take several hours for an A4-sized hologram (20x30 cm).

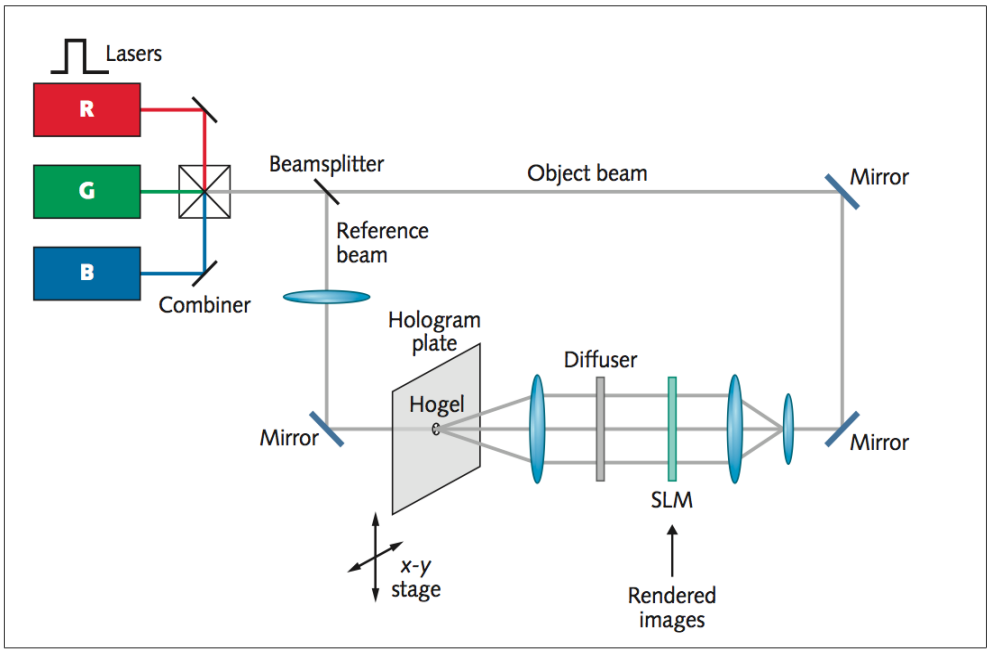

Figure 31: Holoprinter optical architecture

The latest generation holoprinters use nanosecond pulsed lasers to eliminate vibrational sensitivity. Typical RGB laser frequencies are 660, 532 and 439 nm, pulse lengths are typically 35-50 ns and energies are around 5 mJ per pulse. These figures are typical for silver halide materials whereas photopolymers require more energy via multiple pulses. The laser spot size determines the size of the holopixel and the optics ensure that all RGB spots are spatially matched. The size of the holopixel is typically one square millimetre, although the diameter can vary between 0.25 mm to 1.6 mm. Each holopixel is written with a single pulse per colour.

The holographic media require development after the writing process, where silver

halide is developed chemically and photopolymer is developed by exposure to UV. The holographic film is typically laminated onto a hard plastic backing for added rigidity. Some holoprinters print directly on to a glass substrate which is then used as a master for contact copying for large-scale production.

3.1.2 Holopixel synthesis

Each holopixel contains information about the whole image from its own unique perspective. The information for each holopixel is determined by Fourier transforming a series of rendered images generated via computer graphics. These fringe patterns are written into the holographic plate (Figure 31). The Fourier Transform operation may be performed by depicting each holopixel scene on the microdisplay and imaging this via a Fourier transforming lens configuration (Figure 32) where the holographic plate is placed at the Fourier plane (focal point) of the lens. An alternative configuration is numerical computation of the Fourier transform and recording this directly onto the holographic plate via a lens acting in imaging mode. The detailed mathematics and algorithmic details behind the holoprinter image processing is beyond the scope of this section. Instead, this section describes the general principles behind the approach and highlights the application of the spatial multiplexing technique.

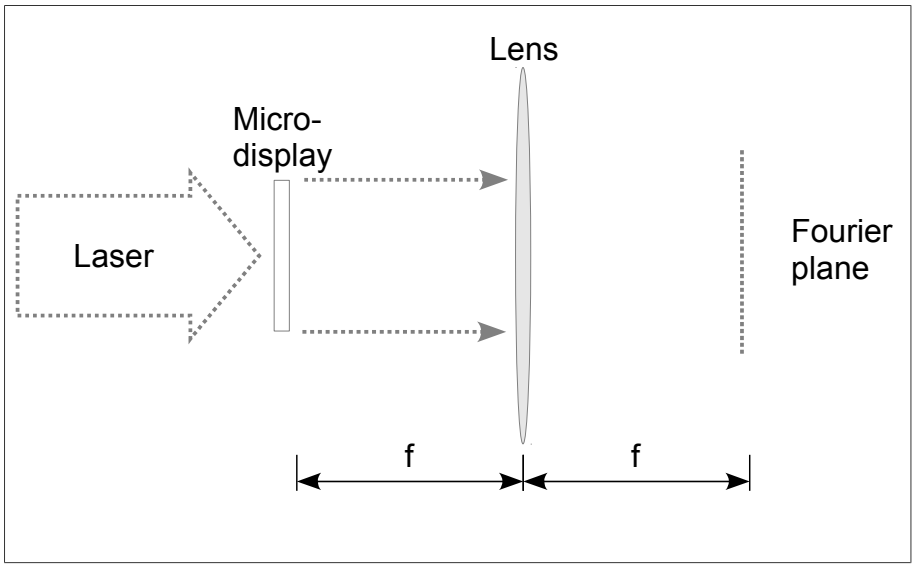

Figure 32: Thins lens Fourier transform

On playback, when the digital hologram is illuminated by a white point light source, each holopixel makes a small contribution to the overall image by optically performing an inverse transform operation. The digital hologram can be envisaged as a matrix of

tiny Bragg-reflectors or micro-mirrors (section 1.3.2) each performing the following functions: (i) filtering the light into red, green and blue components, (ii) reflecting the filtered light back in different directions per holopixel and (iii) the sum of all reflected light contributes towards reconstruction of the scene.

Digital holograms can be either horizontal parallax or dual (horizontal/vertical) parallax. Horizontal parallax digital holograms have a wide horizontal field of view (typically 110 degrees) and a limited vertical field (around 12 degrees); which is the case for Geola's holograms. Full parallax digital holograms have both horizontal and vertical viewability with a field of view of between 50 to 90 degrees, depending on the holoprinter device and the diffraction efficiency of the holographic material; Zebra Imaging's holograms have a 90 degree field of view. The advantage of horizontal parallax holograms is that they require relatively little computation to generate the rendered images, typically several hundreds up to a thousand frames. These are well-suited for larger-area holograms. Full parallax holograms require up to a hundred thousand rendered images, thus increasing the complexity by several orders of magnitude. These are therefore suited for smaller-area holograms.

The holopixels contain patterns produced by the interference of the respective laser beams through the substrate. Each holopixel consists of a diffraction pattern synthesised from a series of rendered frames of the 3D object with a set of views from a series of angles. These frames can be rendered from a 3D computer graphics model using standard 3D tools like 3D Studio Max.

The method is illustrated for four holograms: A, B, C and D:

1. Render Frames[1..N]
2. Compute digital hologram interference patterns H comprising holopixels(x,y)
3. Repeat for Red, Green & Blue
4. Output interference patterns to holoprinter

The rendering and holopixel generation process is illustrated in Figure 33:

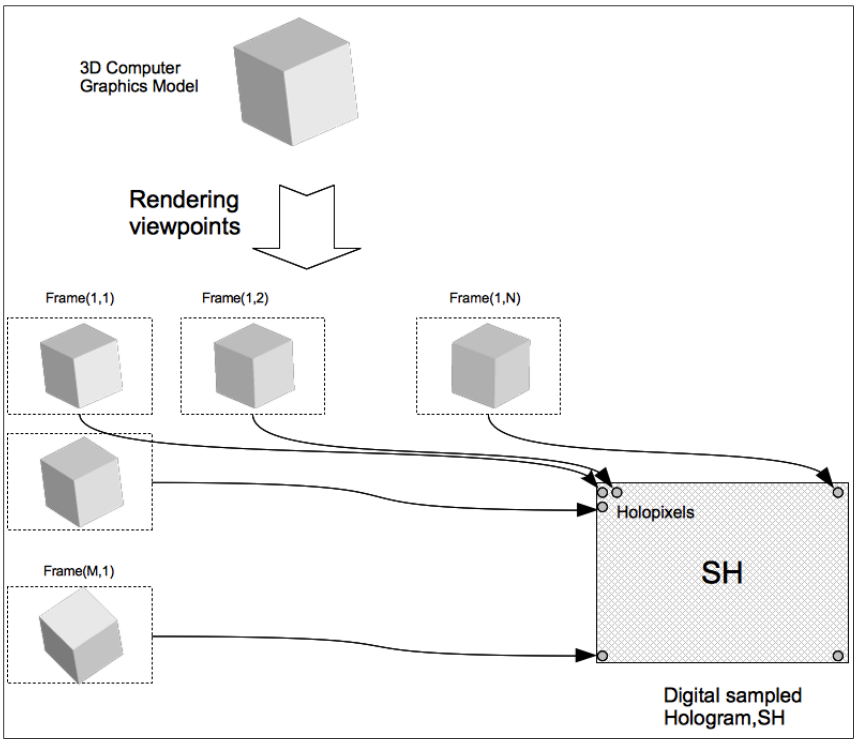

Figure 33: Holopixel rendering and mapping overview

The configuration is shown for full-parallax rendering; single parallax is not discussed here. The digital hologram, *H*, contains MxN holopixels *h(m,n)* which can be written in matrix form as follows:

$$H = \begin{bmatrix} h(1,1) & h(2,1) & \cdots & h(M-1,1) & h(M,1) \\ h(1,2) & h(2,2) & \cdots & h(M-1,2) & h(M,2) \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ h(1,N-1) & h(2,N-1) & \cdots & h(M-1,N-1) & h(M,N-1) \\ h(1,N) & h(2,N) & \cdots & h(M-1,N) & h(M,N) \end{bmatrix} \quad (12)$$

Each holopixel *h(m,n)* contains a processed version of the rendered scene as viewed from its own aperture :

$$h(m,n) = F\left[frame(m,n)_{x,y} \right] \quad (13)$$

In the simplest case, the number of rendered images match the holopixel resolution (M,N), in which case the computation function, *F*, is the Fourier Transform of the scene computed either optically or numerically (section 3.1.2) for recording into the holographic medium. However, in general, there is a mismatch between the viewing position, diffraction capability of the holopixel, holopixel positions and resolution of the rendered images. There exist a host of image and signal processing algorithms that perform the necessary transformations between these various planes. An example

includes the Reconfigurable Image Projection algorithm [83], or RIP, developed by MIT. One of the simplest such algorithms is developed by Geola, known as the Pixel Swap Algorithm which, although computationally straightforward, requires large amounts of memory to hold all of the rendered images in memory at the same time [84,85, p. 293]. The function *F* may additionally include corrections for any optical aberrations inherent in the recording process [85, p. 283].

Reconstruction consists of a white light source illuminating the digital hologram in the usual reflection manner (section 1.3.2). The RGB recording nature implies that each pixel takes the light, performs RGB filtering and reflects these back into space. Each holopixel reconstructs the image from its own unique viewpoint and the sum of these contributions contribute towards the final image.

The holoprinter used for this work is designed and made by View Holographics Ltd, based in North Wales. The work was conducted in close collaboration with this company within the context of a Technology Strategy Board funded research project. View Holographics' holoprinter is based on technology licensed from Geola.

3.2 Digital Hologram Sampling and Interleaving

The main idea behind this research is to construct a holographic screen containing multiple independent sampled 3D images. This means making a sampled and interleaved digital hologram of the cube and sphere shapes similar to those shown in Figure 20. To achieve this, we need to sample and interleave a pair of digital holograms of a cube and a sphere respectively as shown in Figure 34 below.

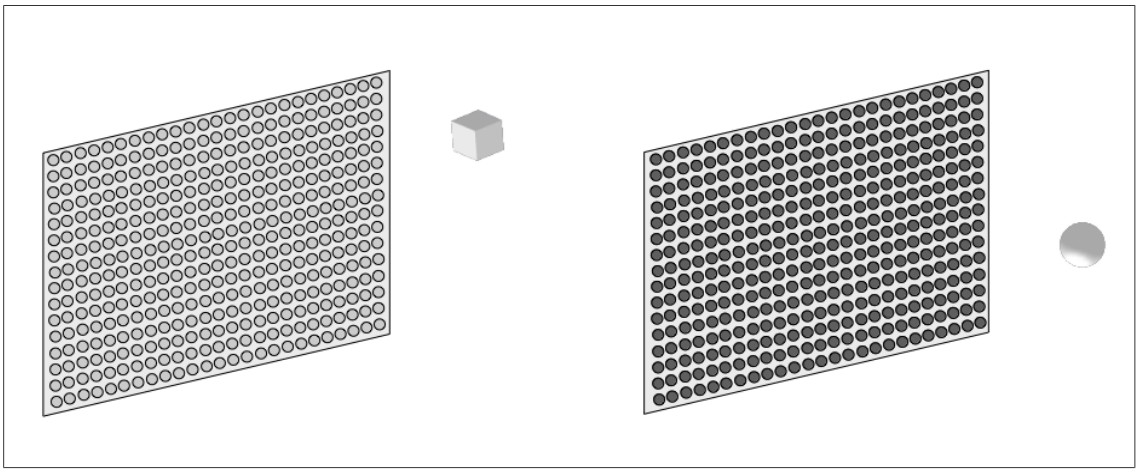

Figure 34: Digital holograms of a cube and sphere shape

A 320x240 mm digital hologram would contain a matrix of 320x240 or 78600 holopixels (M=320, N=240) for a holopixel size of 1.0 mm. The matrix of holopixels

a*(x,y)* for a cube shape and *b(x,y)* can be represented by the corresponding digital holograms, *A* and *B* may be written as:

$$A = \begin{bmatrix} a(1,1) & a(2,1) & \cdots & a(319,1) & a(320,1) \\ a(1,2) & a(2,2) & \cdots & a(319,2) & a(320,2) \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ a(1,239) & a(2,239) & \cdots & a(319,239) & a(320,239) \\ a(1,240) & a(2,240) & \cdots & a(319,240) & a(320,240) \end{bmatrix}$$

$$B = \begin{bmatrix} b(1,1) & b(2,1) & \cdots & b(319,1) & b(320,1) \\ b(1,2) & b(2,2) & \cdots & b(319,2) & b(320,2) \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ b(1,239) & b(2,239) & \cdots & b(319,239) & b(320,239) \\ b(1,240) & b(2,240) & \cdots & b(319,240) & b(320,240) \end{bmatrix} \quad (14)$$

Figure 35 shows how these digital holograms can be sampled and interleaved simply by taking every other holopixel from *A* and *B* using a checkerboard scheme. Figure 35 can be compared to the analogue holoscreen from Figure 20, section 2.2, where the similarity between the analogue and digital approaches is apparent.

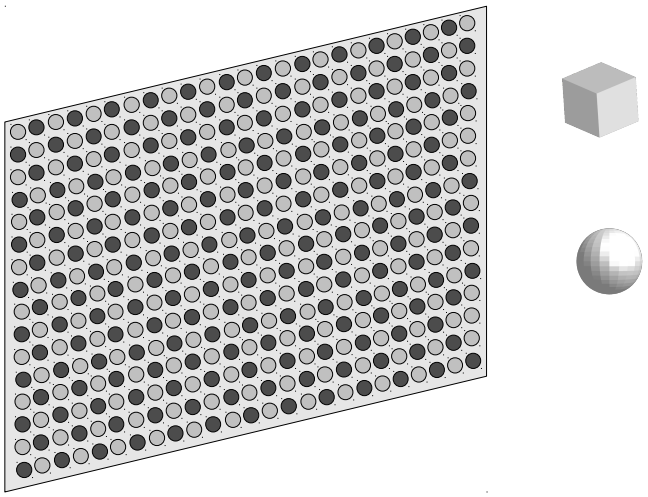

Figure 35: Simple sampled and interleaved hologram

This sampled and interleaved version can be written as:

$$AB = \begin{bmatrix} a(1,1) & b(2,1) & \cdots & a(319,1) & b(320,1) \\ b(1,2) & a(2,2) & \cdots & b(319,2) & a(320,2) \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ a(1,239) & b(2,239) & \cdots & a(319,239) & b(320,239) \\ b(1,240) & a(2,240) & \cdots & b(319,240) & a(320,240) \end{bmatrix} \quad (15)$$

We can show that, for large 3D objects, digital holograms may be sampled and still be able to accurately reconstruct the object.

3.2.1 Analytic overview

The digital hologram of the cube shape, L_C, can be sampled simply by replacing every other holopixel with a zero. Mathematically, this can be expressed as the Hadamard product (element by element multiplication) by the sampling matrix, S and its complement, S' as shown in equation (16) below.

$$S_1 = \begin{bmatrix} 1 & 0 & \cdots & 1 & 0 \\ 0 & 1 & \cdots & 0 & 1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 1 & 0 & \cdots & 1 & 0 \\ 0 & 1 & \cdots & 0 & 1 \end{bmatrix} \qquad S_2 = \begin{bmatrix} 0 & 1 & \cdots & 0 & 1 \\ 1 & 0 & \cdots & 1 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 1 & \cdots & 0 & 1 \\ 1 & 0 & \cdots & 1 & 0 \end{bmatrix} \tag{16}$$

It can be seen that these sampling matrices are composed of the sub-matrices $s_1 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ and its inverse $s_2 = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$

Hence the sampled and interleaved digital hologram from equation (15) is given by:

$$AB = A \circ S_1 + B \circ S_2 = \begin{bmatrix} a(1,1) & b(2,1) & \cdots & a(319,1) & b(320,1) \\ b(1,2) & a(2,2) & \cdots & b(319,2) & a(320,2) \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ a(1,239) & b(2,239) & \cdots & a(319,239) & b(320,239) \\ b(1,240) & a(2,240) & \cdots & b(319,240) & a(320,240) \end{bmatrix} \tag{17}$$

Where the dot operator is the Hadamard product, or element-wise multiplication of the operands.

3.2.1.1 Higher order sampling

The previous examples illustrated sampling and interleaving of two holographic images. The concept can be extended further for higher order sampling. This can be achieved by increasing the number and dimensions of the sub-matrices. For example, a 2x2 sub-matrix arrangement can support four images as follows:

$$s_1 = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \quad s_2 = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \quad s_3 = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix} \quad s_4 = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} \tag{18}$$

$$S_1 = \begin{bmatrix} 1 & 0 & \cdots & 1 & 0 \\ 0 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 1 & 0 & \cdots & 1 & 0 \\ 0 & 0 & \cdots & 0 & 0 \end{bmatrix} \quad S_2 = \begin{bmatrix} 0 & 1 & \cdots & 0 & 1 \\ 0 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 1 & \cdots & 0 & 1 \\ 0 & 0 & \cdots & 0 & 0 \end{bmatrix} \quad S_3 = \begin{bmatrix} 0 & 0 & \cdots & 0 & 0 \\ 1 & 0 & \cdots & 1 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & 0 \\ 1 & 0 & \cdots & 1 & 0 \end{bmatrix} \quad S_4 = \begin{bmatrix} 0 & 0 & \cdots & 0 & 0 \\ 0 & 1 & \cdots & 0 & 1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 1 & \cdots & 0 & 0 \\ 0 & 1 & \cdots & 0 & 1 \end{bmatrix} \tag{19}$$

If *A, B, C* and *D* are different digital holograms corresponding to holopixels *a(x,y)* .. *d(x,y)* then a sampled composite digital hologram can be made by sampling by *S1*, *S2*, *S3* and *S4* to yield the sampled holograms *A∘S1, B∘S2, C∘S3, D∘S4*:

$$A \circ S_1 = \begin{bmatrix} a(1,1) & a(2,1) & \cdots & a(319,1) & a(320,1) \\ a(1,2) & a(2,2) & \cdots & a(319,2) & a(320,2) \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ a(1,239) & a(2,239) & \cdots & a(319,239) & a(320,239) \\ a(1,240) & a(2,240) & \cdots & a(319,240) & a(320,240) \end{bmatrix} \circ \begin{bmatrix} 1 & 0 & \cdots & 1 & 0 \\ 0 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 1 & 0 & \cdots & 1 & 0 \\ 0 & 0 & \cdots & 0 & 0 \end{bmatrix} \tag{20}$$

$$A \circ S_1 = \begin{bmatrix} a(1,1) & 0 & \cdots & a(319,1) & 0 \\ 0 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ a(1,239) & 0 & \cdots & a(319,239) & 0 \\ 0 & 0 & \cdots & 0 & 0 \end{bmatrix} \tag{21}$$

We repeat the sampling for *B∘S2, C∘S3, D∘S4* and add all the sampled holograms to make the holographic screen, *HS*:

$$HS = A \circ S_1 + B \circ S_2 + C \circ S_3 + D \circ S_4$$

$$HS = \begin{bmatrix} a(1,1) & b(2,1) & \cdots & a(319,1) & b(320,1) \\ c(1,2) & d(2,2) & \cdots & c(319,2) & d(320,2) \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ a(1,239) & b(2,239) & \cdots & a(319,239) & b(320,239) \\ c(1,240) & d(2,240) & \cdots & c(319,240) & d(320,240) \end{bmatrix} \tag{22}$$

Note that any number of alternative sampling strategies may be applied e.g 1x4 sub-matrix:

$$s_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix} \quad s_2 = \begin{bmatrix} 0 & 1 & 0 & 0 \end{bmatrix} \quad s_3 = \begin{bmatrix} 0 & 0 & 1 & 0 \end{bmatrix} \quad s_4 = \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix} \tag{23}$$

Or even a 4x4 sub-matrix arranged as:

$$s_1 = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad s_2 = \begin{bmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad s_3 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{bmatrix} \quad s_4 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix} \tag{24}$$

3.3 Holographic screen generation

This section describes how a holoprinter device is adapted to perform the sampling and interleaving of multiple independent digital holograms to make the holographic screen. The simplest and crudest method would be to render all of the digital holograms

and compute the full digital holograms before applying the sampling. The method is illustrated for the four hologram sampling: A, B, C and D:

1. Render FramesA[1..N], FramesB[1..N], FramesC[1..N], FramesD[1..N]
2. Compute full digital holograms (all holopixels): *A, B, C, D*
3. Apply sampling by discarding 3 out of 4 holopixels using *S1* to *S4*: *A○S1 .. D○S4*
4. Interleave (add) the sampled holograms to make the holographic screen, *HS*
5. Repeat for Red, Green & Blue
6. Output patterns to holoprinter

This method is probably the simplest to implement with the least amount of changes to the current driver. However, it is wasteful of memory storage and computation, taking the most amount of time to generate the holopixel data. It is suited to multiplexing just eight holograms as shown here but would be too cumbersome for larger numbers. A number of optimisations could be performed to improve the speed and efficiency of the process. One such optimisation could be to update the driver to combine steps 2 and 3:

1. Render FramesA[1..N], FramesB[1..N], FramesC[1..N], FramesD[1..N]
2. Compute partial digital holograms: *A○S1, B○S2, C○S3, D○S4* directly
3. Interleave (add) the sampled holograms to make the holographic screen, *HS*
4. Repeat for Red, Green & Blue
5. Output patterns to holoprinter

Furthermore, it would be straightforward to combine steps 2 and 3. This would require less memory and speed up the computation.

3.3.1 3D imagery

It was decided to apply the multiplexing scheme from the previous section to four 3D images arranged in a 2x2 configuration. The images were produced in colour with the aim of providing a variety and challenging set of test patterns. The images were produced as 3D models within 3D Studio Max (version 2011) from Autodesk, a popular all-round 3D design package.

The composite test image comprises a series of geometric shapes overlaid in space with distinct colours. The shapes include cubes, blocks and cylinders in solid and

hollow configurations. The composite test image is depicted in Figure 36. This is subsequently broken down into four component images for interleaving in the next section, Figure 37. The colouring between frames is also distinct. It can be seen that the hollow yellow cylinder in the bottom right corner grows with successive frames. The images also have distinct colours for the objects. The background area is a receding checkerboard mesh for added effect.

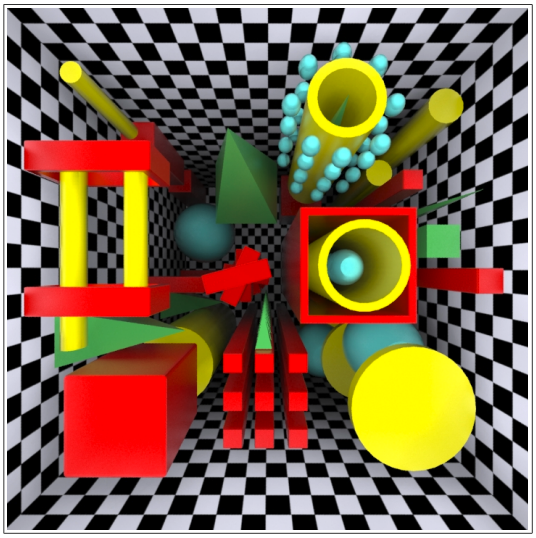

Figure 36: Composite test image

The rendering was in accordance with View Holographics's guidelines. At the time this work was performed, View Holographics's holoprinter was not capable of producing pop-out or real images.

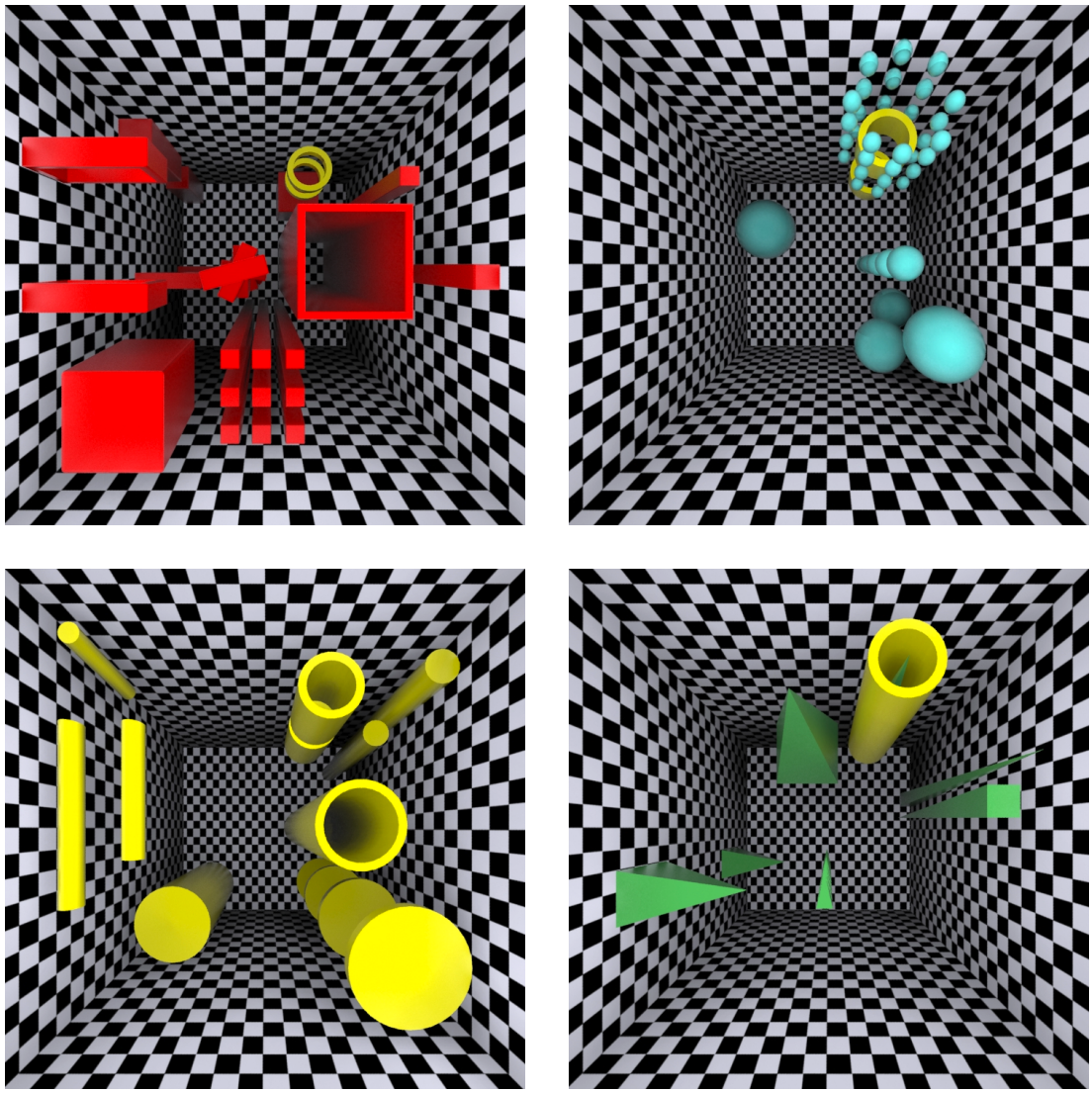

Figure 37: Component test images for interleaving

3.3.2 Holoscreen fabrication

Four sets of rendered images were produced for each of the component images under four directories named *Frame1* .. *Frame4* i.e:

Frame1\
row1\co11.png, col2.png, … col80
row2\co11.png, col2.png, … col80
…
row80\co11.png, col2.png, … col80
Frame2\
row1\co11.png, col2.png, … col80
row2\co11.png, col2.png, … col80
…
row80\co11.png, col2.png, … col80

Frame3\

row1\co11.png, col2.png, … col80

row2\co11.png, col2.png, … col80

…

row80\co11.png, col2.png, … col80

Frame4\

row1\co11.png, col2.png, … col80

row2\co11.png, col2.png, … col80

…

row80\co11.png, col2.png, … col80

A Python program performs the interleaving across the four frames according to the first recipe in section 3.3. In order to keep things simple, it was decided to make an 80x80 mm hologram with 1 mm holopixels, which translates to 360 rendered images. Each image has a resolution of 720x720 pixels. These are stored locally as .png files with lossless compression. The images in Figure 37 are the actual rendered views from a central viewpoint. Each file is approx. 640 Kbytes and the total set is around 2.3 GBytes or around 1 GByte compressed. The data was transmitted to View Holographics via the Internet.

3.4 Holographic Display System

The first generation display was based on a transmission holoscreen with illumination and patterning provided by a laser projector (sections 2.3 and 2.4). Although this configuration gave good results, it has a number of shortcomings of mechanical stability and alignment. The work now splits the illumination and patterning, the former being provided by a simple light source and the patterning is by an optical shutter. The second generation display is based on colour reflection holograms which require a white light point source.

3.4.1 Hardware setup

The initial idea for this phase of the project was to construct a custom optical mask based on a liquid crystal (LC) shuttering device. This would be fabricated as a patterned passive matrix liquid crystal shutter that can either block (e.g. based on twisted nematic LCs) or scatter light (e.g. polymer dispersed or smectic LCs). The shutter mask would be able to recreate any of the patterns necessary for masking the holographic screen (e.g. Appendix A, Figures 87 & 88). Unfortunately, there are very few vendors of custom LC devices left in Europe, so Swedish-based LC-Tec were approached and they

were initially interested in the idea. A suitable display was specified and designed but the manufacturer started to increase the fabrication price, citing the necessity of using high resolution masks for patterning features around 250 microns. This route was not economically viable. Most of the other LC manufacturers are in the Far East and they are not interested in small quantities or R&D projects. In the end, a solution was found in a monochrome active matrix TFT monitor by NEC, which is used for viewing X-Rays in medical imaging, with the following specifications:

- 19-inch display panel (model NL128102BM29-05A)
- SXGA resolution (1280x1024)
- 0.294 mm pixel pitch (0.094 mm sub-pixel pitch)
- 900:1 contrast ratio
- 255 grey levels per pixel/sub-pixel

Each pixel is divided into three sub-pixels, addressed as RGB like a colour display. Indeed, this part is a colour display without the colour filters and the sub-pixels are addressed accordingly. The panel is much larger than required by the project but given the scarcity of mono displays and relative low cost compared to the problematic custom solution, it was decided to go ahead and try it. The backlight on the panel was removed, thus enabling its use as a transparent display. The mono LCD panel requires a special LVDS2 interface to drive it. This is provided by a Fujitsu D2963-S mini-ITX computer with the following specifications:

- 1 GHz AMD Mobile Sempron processor
- Radeon Graphics card with DV-I and dual channel LVDS
- 1 GB RAM
- 4 GB Compact Flash hard disc
- Full I/O (USB, Ethernet, serial etc)

3.4.2 Software and coding

The Fujitsu digital hardware is more powerful than the previous Kontron controller and it can comfortably run Ubuntu Linux (version 10.4). The Python code for mask patten generation and sequencing is extended to provide extra functionality including a grid function for holopixel alignment with the LC shutter mask and a feature to adjust

the aspect ratio of the on-screen pixels. The latter is necessary because it was observed that some commercial LCDs do not have square pixels but rectangular ones, so the pattern generator algorithm needs to take this into account. The code is backward compatible with the first generation display from the previous chapter. The startup command line is the same:

```
> python testpatterns.py default.cfg patterns.seq <-f>
```

The startup menu now includes the new features:

```
Arrow keys: offset (x and y)
< > keys: margin spacings
- + keys: size of tile
[spacebar]: next sequence
s: sequence ON/OFF (timed)
g: grid on/off
k: keystone on/off, a & z: keystone up/down
d: dump configuration parameters to a file
ESC/q: quit
y&u: scaleY +/- 0.005
x&c: scaleX +/- 0.005
```

The code now includes fairly extensive error checking with much improved stability such that it is robust for use in an industrial environment.

3.5 Results and observations

The hardware setup is depicted in Figure 38 showing the transparent LCD and embedded controller board. The holoprinter-based digital holographic screen is behind the LCD and the illumination is not shown.

There were a number of practical problems encountered with the arrangement of the shuttered digital hologram approach. Since digital holograms are reflection holograms, requiring front-illumination, in the first attempt a white light source was placed in front of the assembly at an angle of sixty degrees to the normal. However, this arrangement led to a series of problems. The first obvious issue is the mismatch between the geometries of the LCD pixels and the holopixels of the digital hologram. This leads to a superposition of two disparate grid structures, resulting in a Moiré patterning effect, leading to poor quality of image reconstruction. This problem was partially overcome by careful alignment of the LCD pixels and holopixels and displaying custom mask patterns on the LCD shutter to accommodate the difference in pitch between the two grids. Nevertheless, this mismatch of grids leads to image ghosting and cross-talk

between adjacent frames.

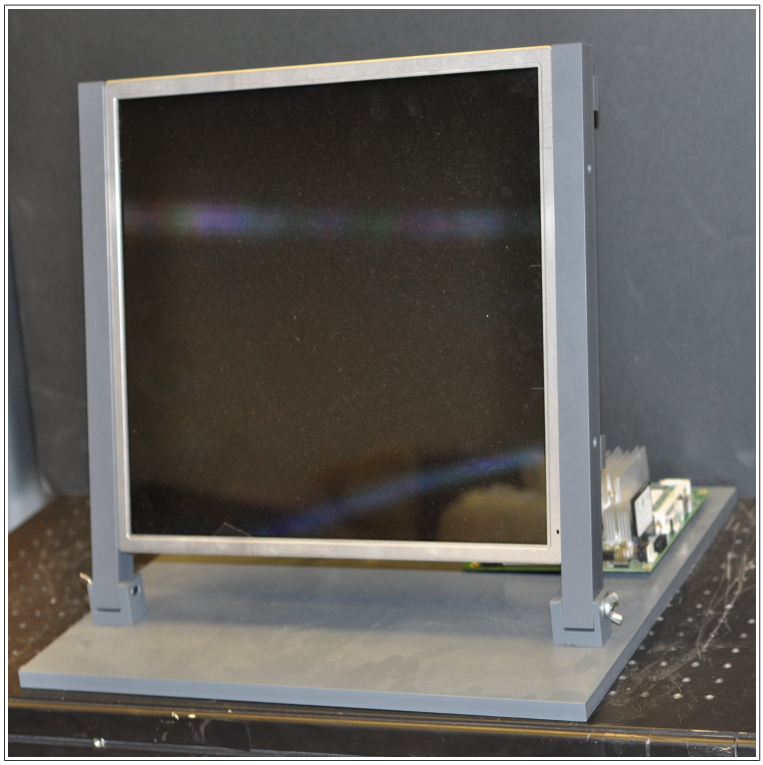

Figure 38: LCD shutter-based holographic display

The second problem centres around the fact that the LCD shutter has a finite width and the digital hologram is mounted on glass with the emulsion on the rear face, Figure 39. Hence there is a relatively "large" distance of around 7 mm between the front face of the assembly and the holopixels. When this sandwich is illuminated at a steep angle, it leads to a shadow masking effect where the LCD shutter pixels cast shadows on the holopixels behind. The shadowing is due to the LCD pixels which do not have a 100% fill factor. The holopixels diffract out horizontally towards the observer back through the LCD shutter. This leads to further reductions in the optical efficacy of hologram reconstruction such that visibility of the reconstructed image was very poor. This problem was partially overcome through the accidental discovery of an alternative illumination method. It was observed that glass-mounted digital holograms can be rear-illuminated with good quality reconstruction. After much thought and deliberation, I realised that this was due to the Fresnel reflection between the incident light and the glass/emulsion-air boundary. When light hits this boundary at the Brewster angle (~56 degrees) [86, pp. 125–159], the s-polarised component gets reflected back. The emulsion sees this as a regular reconstruction beam, albeit a polarised one, thus

reconstructing the image,. The visibility of this rear-illuminated scheme in Figure 40 was qualitatively better than the front-illumination case from Figure 39.

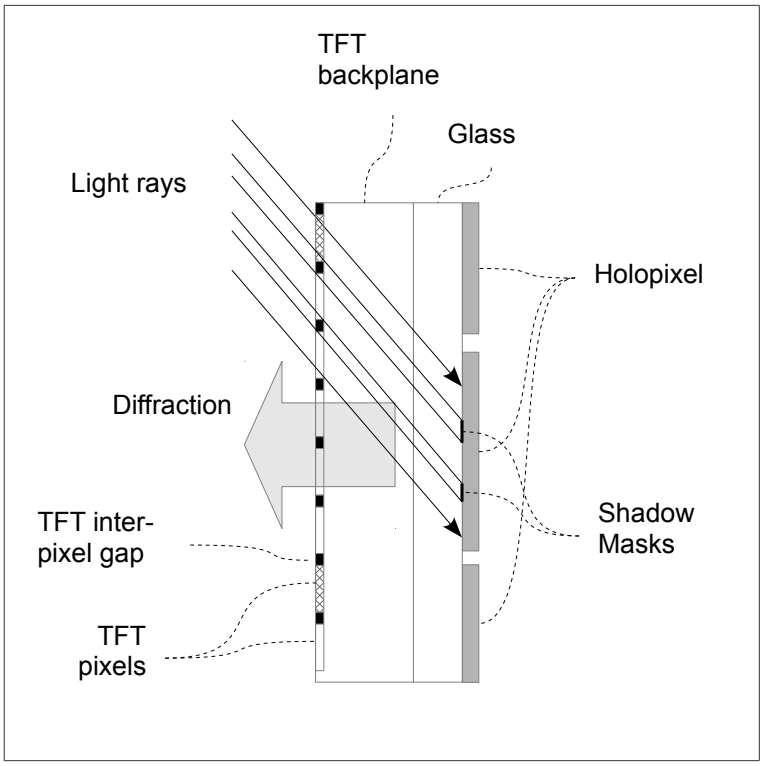

Figure 39: Shadow masking effect

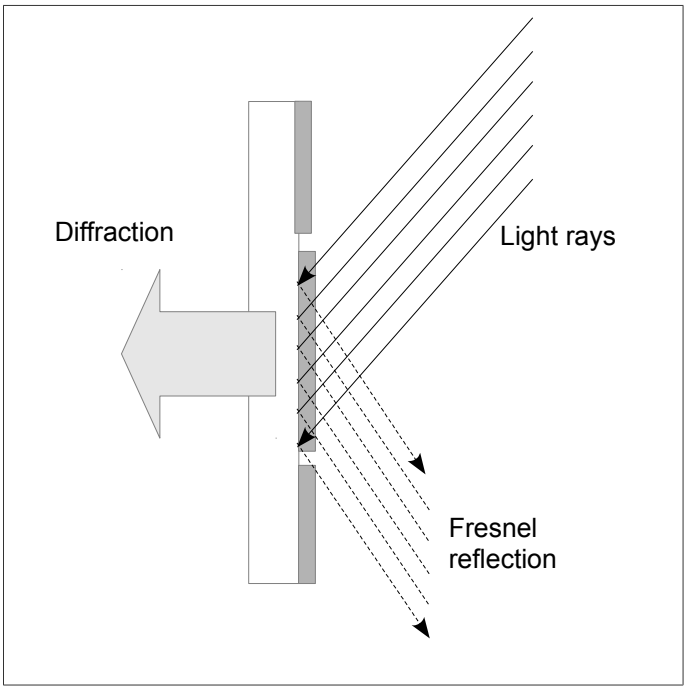

Figure 40: Rear-illumination via Fresnel reflection

As a reference, a composite digital hologram was made of the design from Figure 36, in the normal manner without the sampling, depicted in Figure 41 below:

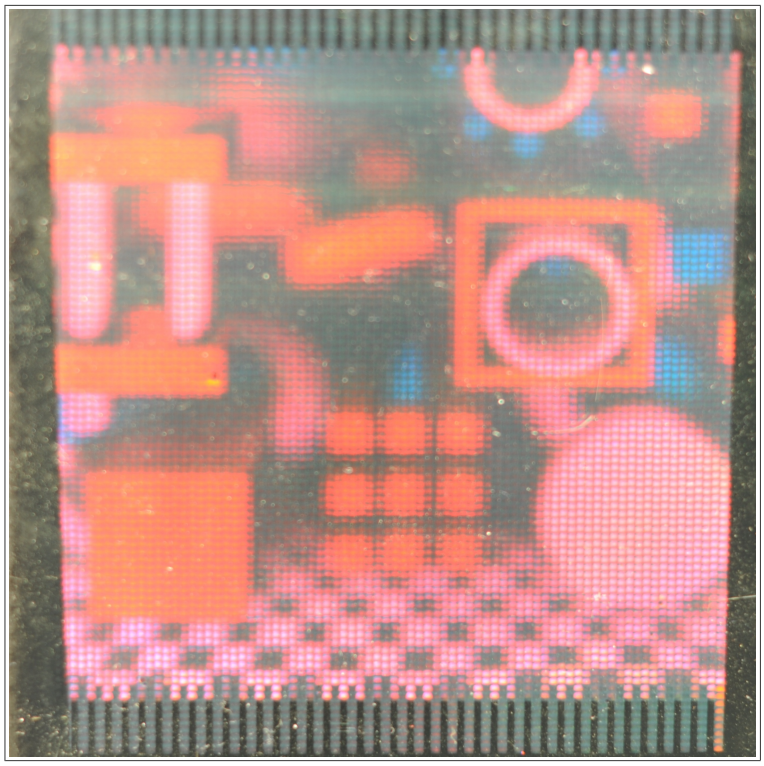

Figure 41: Composite image - all in one

The sampled and interleaved holographic screen is shown in Figure 42, using the rear-illumination scheme, without the LCD shutter. Observe that colour information is lost in this configuration with the red component being most prominent. This is a side-effect of the rear-illumination method, due to a number of factors. The first issue is the wavelength-dependent change in refractive index within the boundary materials leading to different Brewster angles for different wavelengths. The second factor is the four degree angular mismatch between the Brewster angle and the recording/reconstruction beam. Recall from section 1.3.2 that reflection holograms are highly wavelength and angular selective, which reduces optical efficiency.

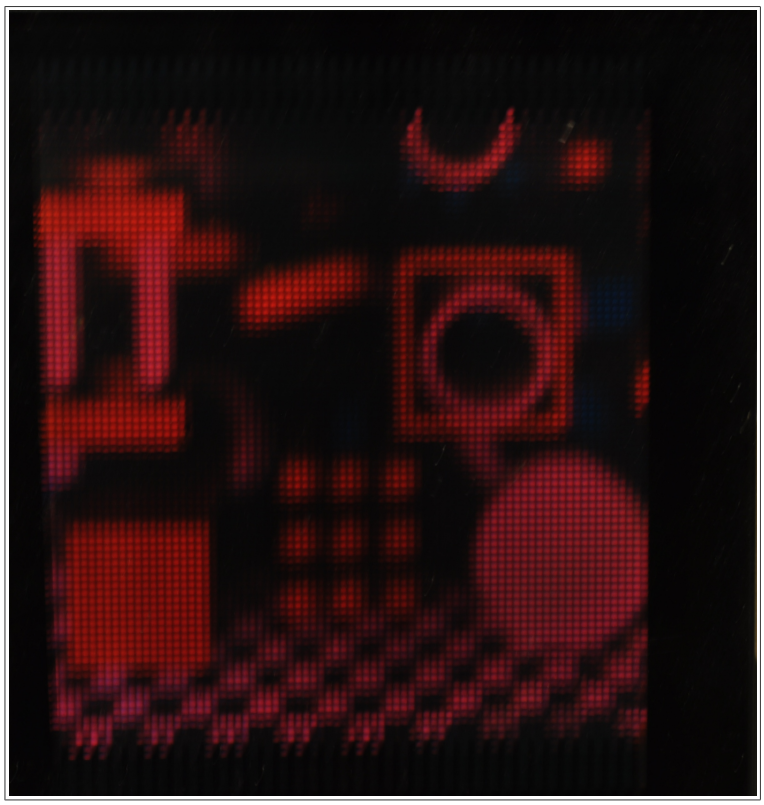

Figure 42: Sampled and interleaved holoscreen

This sampled and interleaved holographic screen was fabricated in accordance with the adapted holoprinter procedures highlighted in section 3.3.2. The holoscreen was then mounted behind the LC shutter, which was configured to switch using mask patterns similar to Appendix A, Figures 87 & 88. A video of the resulting switched sequence was taken and four of the images from this are presented in Figure 43. Although the animation is clearer in the video, some animated sequences are apparent between the top two images whereas the bottom images are difficult to resolve. The main reason for this is the difficulty in aligning the digital holopixels with the LCD pixels. Some image leakage and cross-talk are apparent when comparing the top images. Of the four original component images (Figure 37), only two are clearly discernible, whereas the bottom two images from Figure 43 are difficult to differentiate although it is apparent from the video that something is happening at the transition of the images. The overlapping grids can be seen when Figure 43 is compared to Figure 42, where a periodic hatching structure is apparent in all of the images within Figure 43.

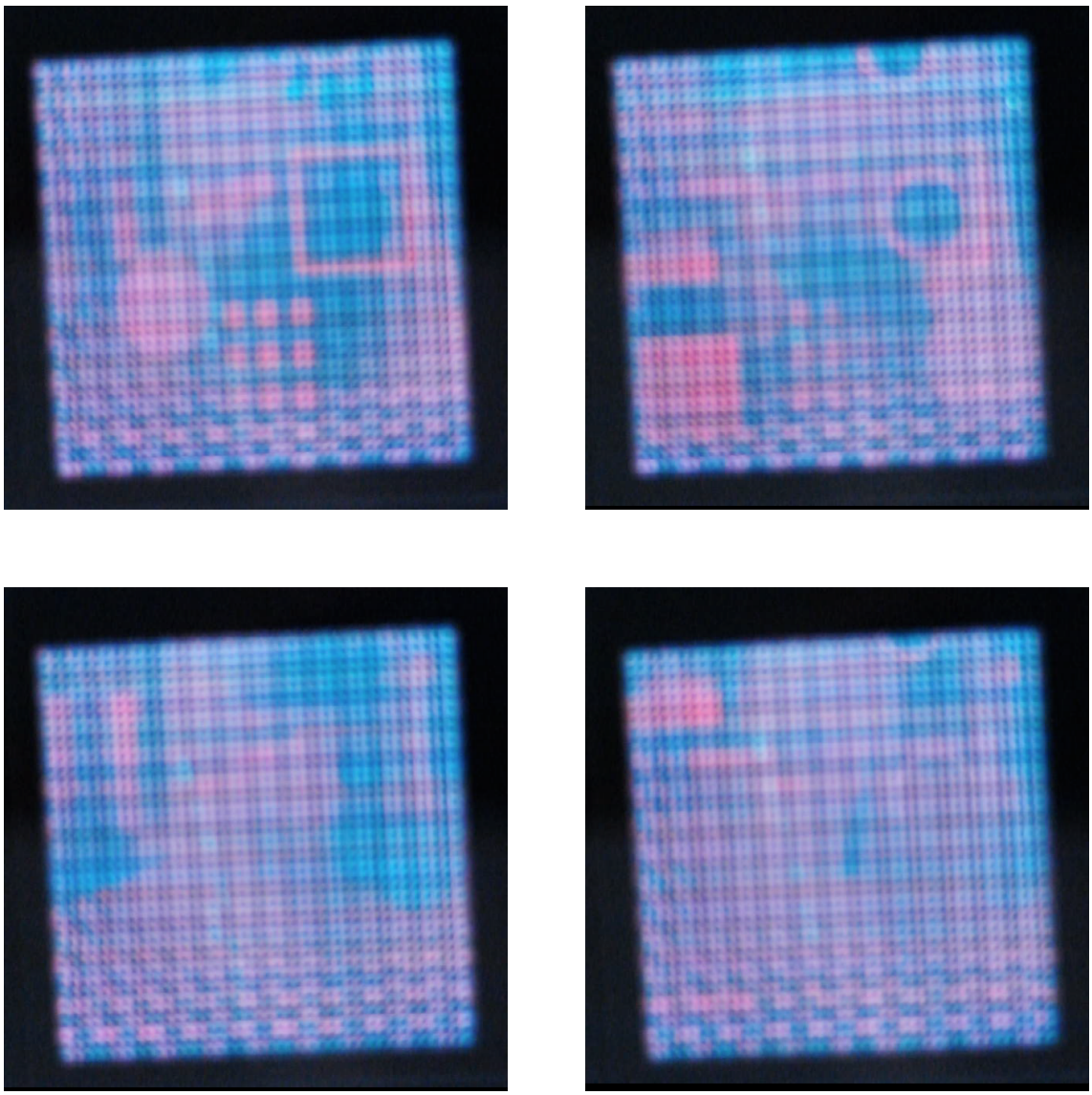

Figure 43: Animation with sampled digital holograms

When the LCD shutter mask is configured as a fully open or 'pass-through' shutter then the image in Figure 44 is observed. This is similar to Figure 42, as expected. This image is highly saturated as it is letting in four times as much light as the individual images from Figure 43, leading to over-exposure on the camera sensor.

As with the work in the previous chapter, all of the images presented here were captured with a Nikon D90 SLR camera. No image processing was performed on the photos, only some cropping.

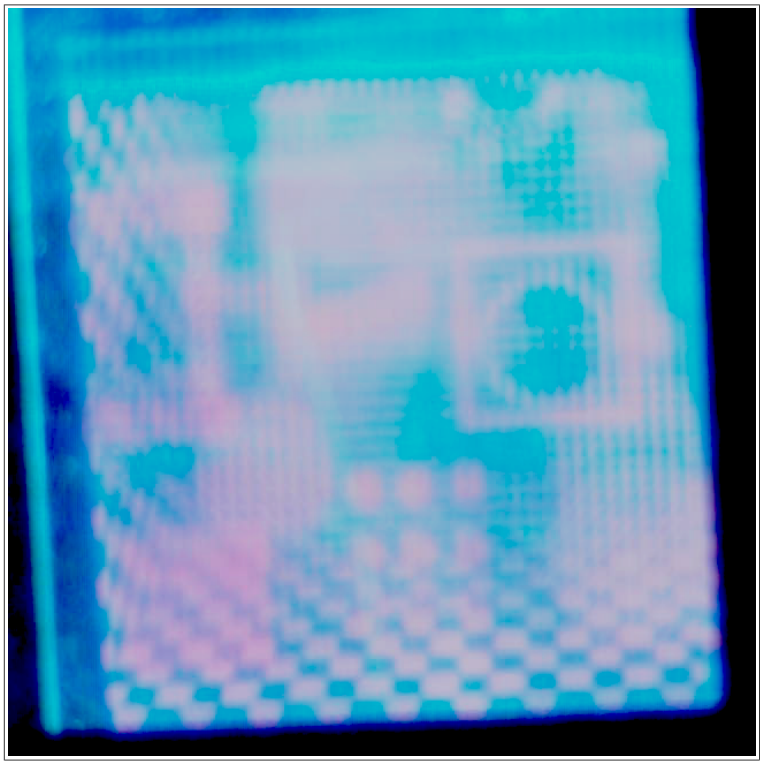

Figure 44: Image with open LC shutter

3.6 Acknowledgements

The 3D models for the imagery in section 3.3.1 were designed and rendered by James Simpson, a 3D graphics designer at Holoxica. The digital holograms were fabricated by View Holographics Ltd. The mechanical design/parts for the display were made by T-Met Engineering and the assembly was done by Mikko Halonen and Craig Dunn at Holoxica. The video for the photos in Figure 43 was taken by Craig Dunn. I would also like to thank the team at View Holographics Ltd, especially Richard Melling for allowing me to reconfigure their prototype holoprinter.

3.7 Conclusion and further work

The work shows that the spatial multiplexing and interleaving approach is valid for reflection holograms made with digital techniques where some crude animation is also possible. Colour is a more difficult challenge and needs further work. I was pleased with the alternative rear-illumination method, where a reflection hologram is recorded at the Brewster angle (~56 deg), enabling rear-illumination via Fresnel reflection. This scheme may have some value for holographers presenting exhibition works.

Although the work shows that spatial interleaving and multiplexing can work with digital holograms, there is more work required to achieve better results. In hindsight, I

think that I tried to tackle too much at once with this work, namely reflection holograms, digital holograms, colour, detailed images and a different illumination architecture. This was perhaps due to a degree of overconfidence or “beginner's luck” given the good results from Chapter 2. A better approach might have been to start with simple monochromatic numeric or segmented shapes, similar to Chapter 2, and build up from there.

As the degree of spatial multiplexing increases, the area occupied by each sampled hologram inevitably decreases. This dilution means that there will be an optical penalty in terms of brightness and quality of reconstruction of the images. If the sampled holograms are of insufficient brightness then this needs to be compensated by a brighter light source in the illumination sub-system. In order to get the best results, it is important to make the most of all of the holopixels available and minimise the number of 'dead' holopixels that make little or no contribution to the hologram (these are often dark source image pixels). This implies large bright objects within the source artwork. The source objects should also be relatively simple in order to preserve quality (i.e. simple texturing, shading etc).

The holopixels in most holoprinters are written with overlaid interference patterns using red, green and blue lasers. As progressive layers of information are recorded within the same physical space, there is a tendency to saturate the holopixel where successive layers cause cross-talk and mutual interference. The cause of this limitation is the resolution of the holographic medium, for silver-halide it is the size of the crystalline grains. Hence, there are limits to the information-storage capacity of a holopixel that are dependent on material properties. The result is that reconstructed images become blurred or hazy. This problem can be overcome by limiting the viewing angle to compensate for the loss of diffraction efficiency, which can be quite significant e.g. dropping from 70 degrees to just 45 degrees. Another way to overcome this limitation is constraining the source image to pure colours or simple combinations of colours with the aim of avoiding saturation. For example, if an image comprises a pure green object then the resulting digital hologram would have a full 70 degree field of view.

3.7.1 Future work

It would be worthwhile reconfiguring the holoprinter recording angle slightly to match the Brewster angle in order to improve the efficiency of the rear-illumination approach. Also, it would be helpful to match the dimensions of the holopixels to those of the LCD

pixels to eliminate the chicken-wire and other undesirable image leakage effects. There is a trend towards even smaller holopixels, down to 0.25mm. It has been shown [87] that holopixel size of 0.4 mm (diameter) or less is almost indistinguishable from an analogue hologram, assuming the source image is well-captured and rendered.

One simple way of scaling back to ordinary reflection holograms is to apply the spatial multiplexing technique from Chapter 2 to record a Denisyuk hologram [51,88, p. 374]. The masking can be performed using a transmissive LCD which changes patterns between exposures, as well as changing the object to be recorded. This is a simple one-step recording process, where the hologram material and masking LCD can be in contact with each other during the recording process. If photopolymer is used then the hologram material does not even have to be removed for processing. This would enable both recording and replay within the same piece of optical hardware. Replay would require a white light source (section 1.3.2).

It would appear from this work and the work in the previous chapter that transmission holograms are better suited to the spatial multiplexing technique when it comes to replaying the stored holograms. This is due to the awkward illumination setup of a reflection hologram as well as the sub-optimal masking via the transmissive LCD. Although holoprinters can, in principle, be reconfigured to produce transmission as well as reflection holograms, they are almost exclusively used in reflection mode. Reconfiguring the device to work in transmission is a major exercise which is not to be taken lightly and manufacturers are even less willing to make this change. Nevertheless, Geola are working on a prototype that produces rainbow transmission holograms for security applications [89]. This may be suitable for the spatial multiplexing approach and will be explored in the near future.

The digital hologram methods described in section 3.1 do not actually reconstruct the wavefront of the image, which is instead approximated through summing a series of tiny viewpoints. Although this method is used in industrial-scale machines to make full-colour full-parallax 3D images, the depth of field of the images is limited to roughly one-third of the hologram diagonal, as a rule of thumb. Recent work by Nihon University shows a holoprinter architecture that can print volume holograms, based on CGH wavefront coding [90]. This approach does not require multiple viewpoints, instead the 3D image is decomposed into a collection of discrete points. This method is an interesting variant on CGH approaches discussed in the next chapter, and it is also amenable to purely digital spatial multiplexing and interleaving explored in Chapter 7.

Chapter 4 - Computer Generated Holography (CGH)

Chapter 2 showed that a simple sampled and interleaved holographic display could be fabricated using variations on classical analogue holographic techniques i.e. holograms fabricated in transmission mode with real objects. This fabrication technique is not easily scalable for more volume segments because it is manual and relatively labour-intensive. Chapter 3 extended the notion to reflection holograms using an automated method via modifications to holoprinting technology and introduced a virtual object represented digitally using computer graphics. The holoprinting method is economically viable but requires further research and close collaboration with the manufacturer to improve performance. Holoprinters are large and expensive machines with extensive background intellectual property and decades of prior research and development. They are used for industrial-scale production so the manufacturer is not keen to take the machine offline for R&D. Prototype machines are not always in good shape for research or experimentation as these tend to be cannibalised for spare parts. There are only a handful of holoprinter manufacturers (section 3.1) which makes it a small community to choose from; although we have good relations with them.

The work from Chapters 2 and 3 shows that currently, transmission holograms are much brighter than reflection holograms, and are easier to work with because they require rear-illumination which leads to a more straightforward optical architecture for shuttering than front-illumination. In the previous chapters, the holographic screen was produced externally and the hologram itself is always analogue, ultimately relying on the interference of a pair of laser beams in a holographic material. The work here focuses on the direct fabrication of holograms using computer-generated interference patterns with visualisation on a Spatial Light Modulator (SLM) and fabrication via lithography as Diffractive Optical Elements (DOE). This could eventually lead to mass production via embossing (section 1.6.3).

In this work, there is full end-to-end control of the hologram and the process is digital all the way, starting from the representation of the object to the fabrication of the hologram. Hence, there is no reliance on third party hologram services or manufacturers, apart from the foundries that produce the DOEs. The work starts with digital Fourier Optics theory that leads to a set of algorithms and design tools to generate digital computer generated holograms, or CGH [35,91]. The holograms can be verified using a spatial light modulator for experimentation and implementation is done via pure phase gratings.

It is important to distinguish between digital holograms and computer-generated holograms at this point, for the avoidance of doubt or confusion, since the literature sometimes uses these terms interchangeably. Digital holograms are produced by holoprinters (section 3.1) which are designed or rendered via computer graphics techniques; however, the holograms are written into the holographic material via laser interference to make reflection holograms. Hence, these are digitally generated but the production is ultimately analogue. On the other hand, we define computer-generated holograms (CGH) in terms of both representing the images digitally, computing the interference patterns and writing these directly on to the holographic material or an appropriate device; in other words, this is a DOE or an SLM. These definitions are consistent with some of the latest literature [53].

4.1 Direct Convolution using Spatial Light Modulator (SLM)

CGH techniques essentially compute the diffraction pattern for a transmission hologram by calculating the complex light field between an object wave and a plane reference wave. The mathematics behind scalar diffraction theory is presented in Appendix C, which is concerned with mapping the field due to a point in the object plane propagated to the hologram plane. The field or interference pattern is governed by the Kirchoff diffraction integral (Appendix C), which is repeated here for convenience:

$$g(x,y;z) = -\frac{1}{j\lambda}\iint_S u_0(x_0,y_0;0)\frac{z\exp(jkr)}{r^2}\,dx_0\,dy_0 \qquad (25)$$

$$\text{where} \quad r^2=(x_0-x)^2+(y_0-y)^2+z^2 \quad \text{and} \quad k=\frac{2\pi}{\lambda}$$

The semi-colon notation for function variables *(x,y;z)* in Equation (25) represent planar continuous variables in the *(x,y)* plane, whereas *z* is a pseudo-constant, thus constraining the equation complexity to just two dimensions. Equation (25) represents the field resulting from the diffraction of an object on the surface *S* (*z=0*) to the hologram plane located at *z*, where *r* is the distance between the points ($x_0,y_0,0$) and (*x,y,z*); and *k* is the wave vector. Equation (25) can be difficult to solve analytically, depending on the profile of the source image, u_0. The integral may be expressed in discrete numeric form by quantising the hologram and object planes. The hologram plane is divided into *M*x*N* holopixels whereas the object, *U*, is decomposed into a discrete set of *P* single-photon emitters across the surface *S*, and the integrals are expanded as summations in expression (26):

$$H[x_n, y_m, 0] = \sum_{m}^{M} \sum_{n}^{N} \sum_{i}^{P} U_i(x,y,z) \frac{z_i \exp[jkr_i]}{r_i^2} \tag{26}$$

4.1.1 Algorithm design and computational requirements

Each point of the hologram plane contains the summation of the complex E-fields of all object points. Each object point is represented by an amplitude value and three spatial co-ordinates. In terms of computation, expression (26) is broken down further by expanding the complex exponential term by Euler's relation (27):

$$H[x_n, y_m, 0] = \sum_{m}^{M} \sum_{n}^{N} \sum_{i}^{P} U_i[x, y, z] \frac{z_i}{r_i^2} [\cos(kr) + j \sin(kr)] \tag{27}$$
$$\text{where} \quad r_i^2 = [x_i - x_n]^2 + [y_i - y_m]^2 + z_i^2$$

In terms of computation, these terms represent fifteen distinct mathematical operators (add/subtract, multiply/divide, square root and trigonometry). The total computation is then given by:

$$T = P \times M \times N \times C \tag{28}$$

Where C is the actual number of computational operations required to implement these depend on the architecture and performance of the underlying computer system. Most modern computers are equipped with floating point co-processors that can concatenate multiplications with additions/subtractions (MAC operations) and perform these in a single CPU cycle. Divisions and trigonometry are performed in 2-3 cycles and the square root in up to ten cycles, depending on implementation. GPUs and DSPs include hardware accelerators that can compute divisions or trigonometry in a single clock cycle; and square roots in a few cycles.

Whereas the analytic expression (25) is difficult to solve, the numerical version (26) is difficult to compute [92]. This can be illustrated using expression (28) in a few examples. If the object is decomposed into a set of single-photon emitters, then the question arises about the spacing of these emitters. The point separation of 50 μm is designed to match the visual acuity of the eye at 1.0 m, i.e. the eye cannot resolve two points spaced this distance apart from a metre away. For example, at the given resolution, a cubic millimetre of space contains 8,000 emitters; if the hologram plane is

a 1 Megapixel SLM. If these parameters are applied according to expression (28) then we require 160G operations per mm^3. The following table presents this scenario together with a square centimetre (10x10mm) containing 40,000 points and a cubic cm with 8 million points; presented for a 1M pixel SLM and a 1G pixel DOE. The figures in the table below assume C=20 operations per computation.

Object size	Object points, P	Hologram size, MxN	Compute Ops, T	Type
1 mm^3	8K	1M	160G	SLM
1 cm^2	40K	1M	800G	SLM
1 cm^3	8M	1G	160T	SLM
1 mm^3	8K	1G	160T	DOE
1 cm^2	40K	1G	800T	DOE
1 cm^3	8M	1G	160P	DOE

Table 1: Computation parameters for various voxel sizes

In order to put some of these into perspective, a cubic mm of space is about the size of a sugar crystal grain and a cubic cm is a sugar cube. A full-populated cubic cm of space is an unlikely scenario but is presented here for comparison. Computations requiring up to ~10T operations can be performed on an ordinary computer, however, anything needing tens of Terra operations implies the use of high performance computing for larger hologram sizes and objects. The initial idea was to use the powerful Edinburgh University supercomputing cluster. After some discussions with the experts, it was decided to attempt implementation on GPUs instead [92,93]. It turns out that the latest crop of Nvidia GPUs are relatively low cost and are aimed at high performance computing.

4.1.2 Algorithm implementation

I implemented the algorithm in Matlab and optimised it for matrix operations before porting it to C and then on to the GPU. The computation was performed on a Mac with a dual-core 2.66 GHz processor, 1GHz system bus and 4 GB RAM. The performance figures are shown in Table 1.

There were some problems with the precision of the Nvidia Geforce 8800 GTX GPU to do the complex arithmetic calculations required by the convolution kernel. The early GPUs were constrained to single precision floating point which restricted their use for my algorithms, although they run fine on CPUs, albeit much slower. The problem is

illustrated in Figure 45 showing the interference pattern due to a single point source; the left plot is the double-precision version with finer detail than the single-precision plot.

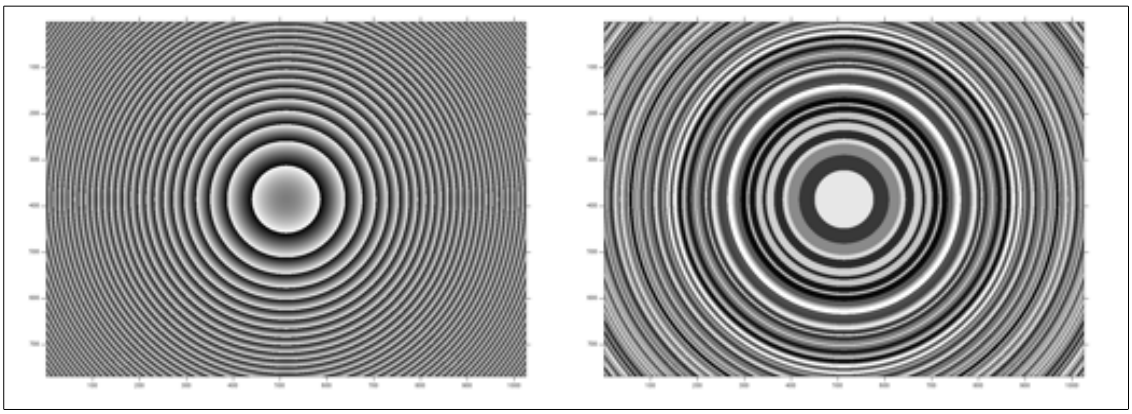

Figure 45: CGH patterns for double precision (left) and single precision (right)

The particle physics group at Edinburgh University kindly let me access one of the EPCC High Performance Computing machines that includes Nvidia Tesla C2050 Fermi GPUs with 448 cores and 500G Flops of double-precision floating point and 3GBytes of RAM. This finally enabled the GPU code to run in its full double-precision glory. The performance results are summarised in the table below.

Nr. Points	**10**	**100**	**1K**	**10K**	**100K**
Matlab – Mac (s)	2	21.1	216.1	4519.1	-
CPU – Fermi (s)	0.98	9.2	90.9	907.0	-
GPU – Fermi (s)	0.0074	0.06	0.53	5.24	52.8

Table 2: CPU vs GPU performance

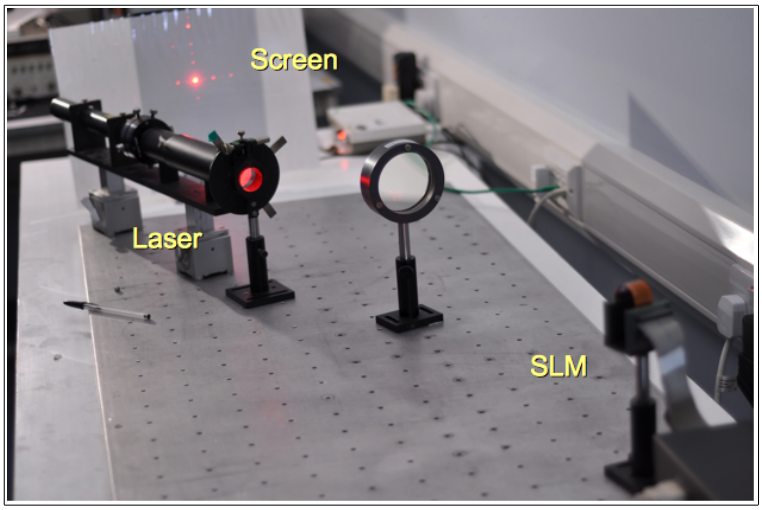

Figure 46: Experimental setup for SLM

It can be seen that the code scales linearly with the number of points in the object – the CPU-bound code takes too long to handle 100k points. The GPU has 171x speedup compared to the CPU code running on the same machine; and the C-code runs twice as fast as Matlab on the Mac.

4.1.3 Experimental results on SLM

The algorithms were tested on a Holoeye LC-R 2500 Spatial Light Modulator[2] (SLM). This is an old part that needed to be resurrected from scratch with the latest drivers and vendor-supplied software. It has a resolution of 1024x768 pixels with a pixel pitch of 19μm. The SLM was illuminated by a 1mW HeNe laser (λ=632.8μm) with collimating optics and a screen, shown in Figure 46. The diffraction angle of the SLM is given by equation (29):

$$\theta = 2\arcsin\left(\frac{\lambda}{\Delta H}\right) \tag{29}$$

where ΔH is the holopixel pitch. For the given parameters, θ = 3.82 degrees, which is a very narrow viewing angle. The images were designed for a distance of 2.5 m from the SLM. The algorithms produce a phase profile between 0 and 2π, which is quantized to 256 levels for the SLM. The input images consisted of either mathematical shapes (circles, squares, lines) or collections of points obtained from bitmap images. Some of the generated images are depicted in Figure 47.

2 Holoeye LC-R 2500 datasheet

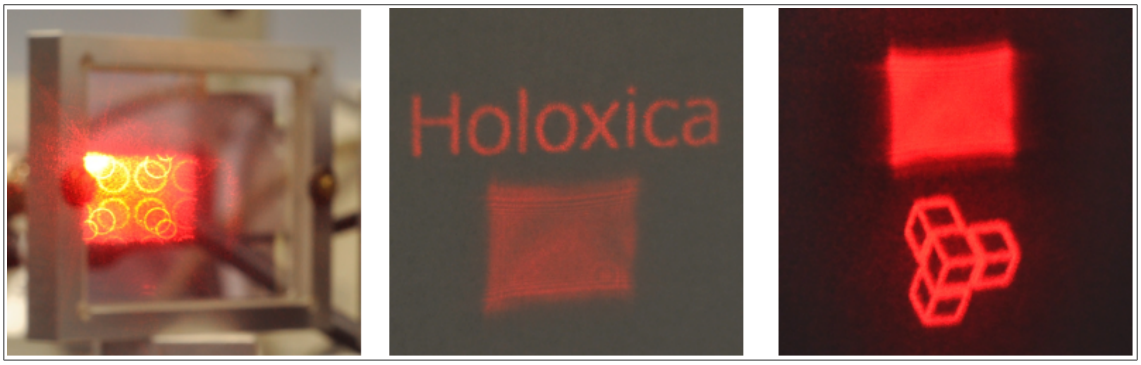

Figure 47: SLM image results

Unfortunately, the SLM is a reflective-mode device so it is not possible to observe near-surface shapes very easily because of the glare from its front surface. A transmissive SLM would have been better in this case. Nevertheless, the exercise has given me confidence in the validity of the algorithms. The code could produce holograms as good as the vendor-supplied software that is probably based on Iterative Fourier Transform methods.

4.2 Diffractive Optical Elements (DOEs)

A basic introduction to DOEs was given in section 1.4.2. The GPU algorithms produce a phase profile of between 0 and 2π, for each holopixel, which is then quantized to 256 levels for the SLM. For DOE fabrication, four or eight levels are required for good (81-95%) diffraction efficiency for a phase-only element [57]. Some initial exploration on quantisation was also done on the SLM. There are very few places in the UK that can make DOEs. After looking around for a suitable foundry to do the work, I gave up and decided to do it at the Scottish Microelectronics Centre (SMC) instead.

The DOE is designed for lithographic fabrication [94] on a 4-inch (10 cm) transparent BK7 (Pyrex) wafer. The holopixel dimension is two microns, which gives a diffraction angle of around 40 degrees; much better than the SLM. The DOE will be fabricated by depositing an oxide layer that is patterned by a photomask and dry-etched (reactive ion). This work was done under subcontract by the SMC.

The GPU-based tools are used to generate the photomasks. A number of DOE holograms were designed. The first one is a simple 2-micron test wafer with a standard checkerboard pattern used for measurements (uniformity, refractive index etc) to characterise the process. The idea is to fabricate a benchmark hologram of a square, based on a binary implementation, followed by a 4-level hologram.

4.2.1 DOE Design Workflow

The starting point of the DOE work on the CGH algorithms is based on direct convolution written in Matlab for SLMs from the previous section. This was extended to making diffractive optical elements using lithography. The basic algorithms are rewritten in C++/CUDA (for GPU supercomputing) that generate a phase profile. A new phase mask tool was written from scratch in C++ to take the phase profile, perform quantization and generate a *gds2* layout (text-based) file for the photomask. The gds2 file was imported into a freeware mask design tool, *Klayout*, where it could be further processed and saved as a more compact binary. An overview of the design workflow across the various tools is presented in Figure 48.

The holographic image design is a reference benchmark square shape 22x22 mm with 48,000 points at a distance of 35mm from the hologram plane. The diffractive element dimensions are 50x50 mm (square) on the wafer with 625M pixels (at 2 μm). This is represented as a point cloud dataset comprising [x,y,z] co-ordinates and amplitude information (.pdat).

The GPU algorithms take about an hour to generate a phase profile (25 MBytes) using a simple internal binary format (.fdat). The mask generation tool, *phase2im*, converts this into a gds text file (20.1 GB), which is then compacted using *Klayout* (9.4 GB) by converting it into a binary gds file that is further gzipped (1.7 GB). It takes about half a day to run a design through the workflow. The resulting file is supplied to the mask vendor, Compugraphics, on a DVD-ROM.

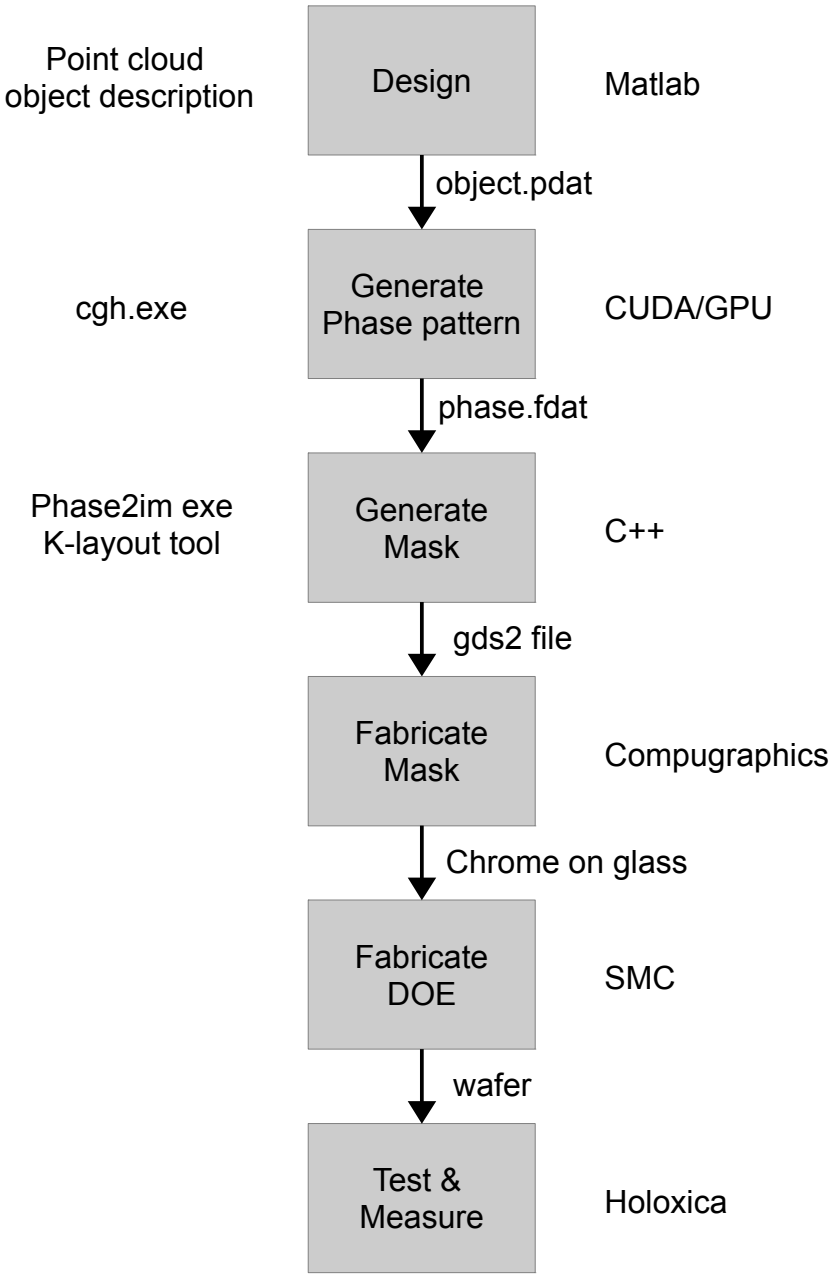

Figure 48: DOE design & fabrication workflow

4.2.1.1 Data file formats

The *.pdat* {point data} file format is used to hold the point cloud data as a set of [x,y,z] co-ordinates plus an amplitude value. It is a simple binary format of single-

precision floating point numbers arranged as follows:

```
M,N,  a1,a2,..,aN, x1,y1,z1, ax2,y2,z2, … aN,xN,yN,zN
```

where *M* is the total number of points and *N*=3. This is a simple intermediate file format designed to inter-operate between various different programming languages. So, *.pdat* files can be written/read by C/C++ and Python programs. This format is used mainly by Matlab to generate objects for processing by other programs. The Matlab programs are: *readpointcloud.m* and *writepointcloud.m*

The .fdat {float data} file format is used to hold scalar matrix data as an MxN matrix of single-precision floating point values.

```
M,N, D[1,1],...D[1,M], D[2,1],D[2,2],..D[2,M],..., D[N,1],D[N,2],..D[M,N]
```

Again, this is a simple matrix format designed to inter-operate between different languages. This is used to transfer amplitude or phase information between programs. The Matlab programs are: *writefloatdata.m* and *readfloatdata.m*

4.2.1.2 Point cloud generator

The generator also produces a .info file which passes design parameters to the DOE phase generator program (next section). This is a simple text file which looks like:

```
points.info: <text file>

points=2000
Z=3.004748e-01
pitch=2.375000e-06
lambda=6.328000e-07
topleftX=-1.500000e-02
topleftY=1.500000e-02
blocks=790
sym=0
```

This example shows a 2000-point object at a distance of 0.3 m from the hologram plane, with a holopixel pitch of 2.375 μm (a sub-multiple of 19 μm) at a wavelength of 632.8 nm. The *topleft* parameters specify absolute co-ordinates and the *blocks* parameter is the number of GPU block units required for computation. The *sym* parameter specifies a symmetric hologram, which can be used to optionally optimise subsequent processing.

A 12.1K point *pdat* file is 193K bytes.

4.2.1.3 DOE generator and phase quantisation

The DOE program is written in C++/CUDA and it performs a direct convolution between the point cloud and the hologram plane. It takes a *.pdat* file and outputs a phase profile as a *.fdat* file. It is invoked by:

```
> ./doe objects/points.pdat outdir
```

which takes the points from a directory objects/, performs the convolution and outputs the resulting *points.fdat* file into the directory *outdir* as an *fdat* file (see previous section). If the *sym* flag is set, then the object and hologram planes are symmetric, which means that only a quarter of the hologram plane needs to be computed. Also, each point is mirrored around the x and y axes, effectively quadrupling the number of points thus ensuring a perfectly symmetric input.

The *phase2im* program takes the *fdat* phase profile from the previous program and performs quantisation before converting the profile into a *.gds2* text file. If the sym parameter is set then the appropriate transformations are to expand the single computed quadrant into the other three quadrants, thus expanding it by fourfold. The program can also generate a bitmap file for presentation on to an SLM, by performing clipping and sub-sampling (see section 4.2.4) on the phase profile.

4.2.1.4 K-Layout tool

K-Layout is a free open-source viewing and editing tool for chip design and layout. It runs on all platforms including Linux, Mac and Windows. The primary use of this tool is to take the text-based gds output from the previous tool and convert it to binary gds for a more compact representation. It is also possible to use this tool to manually add features such as logos or text to the layout.

4.2.2 Photolithographic DOE fabrication

The phase plate is fabricated on a 4” (100 mm) Pyrex wafer with feature sizes of 2 μm. A simple test mask comprising of a grid-style binary grating was designed by the SMC using Clewin, a commercial mask design tool. The photomasks are made by Compugraphics. The SMC tried a number of methods to fabricate the test wafer, as they do not have a great deal of experience with optical fabrication. The active layer is a transparent PECVD (Plasma Enhanced Chemical Vapour Deposition) oxide film deposited on Pyrex as well as silcon wafers for control purposes. The SMC conducted various tries including a number of different patterning and etching methods using

various chemicals and steps. They tried to optimise the quality in terms of the lateral features, uniformity and etch depth. The fabrication steps are:

1. Deposit PECVD oxide on Pyrex wafer
2. Deposit aluminium layer (as a photo-stop for patterning)
3. Spin photoresist
4. Apply photomask and expose/pattern resist
5. Etch metal, etch oxide till the pyrex (etch stop)
6. Strip photoresist and strip metal layer.

These steps are for a binary DOE. More levels require repetition of these steps.

4.2.3 Results and observations

The two micron binary test pattern wafer was fabricated with the mask covering the entire surface area, Figure 49.

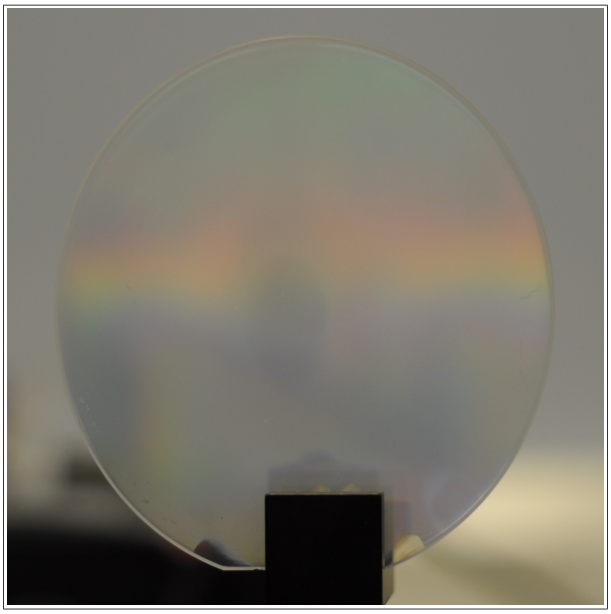

Figure 49: Fabricated binary test wafer

The wafer was viewed under a microscope at the centre and edges, Figure 50. The leftmost image from the centre shows circular etched patterns instead of square ones. The mark/space ratio is also not uniform with the highlighted features at 2.15 μm and 1.85 μm. There is also some considerable over-etching further away from the centre. The right-hand edge view shows squarer pixels but the mark/space ratio is also non-uniform at 1.90 μm and 1.74 μm.

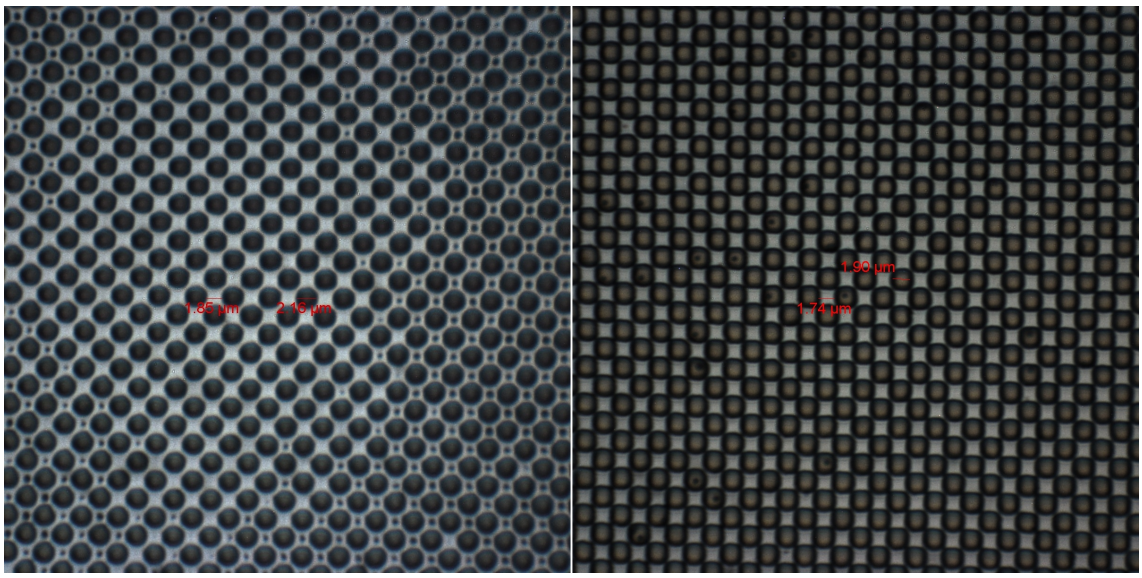

Figure 50: Test wafer measurements at centre (left) and edge (right)

The optical characterisation was done by illuminating the test wafer with a laser and observing the resulting diffraction pattern on a screen, Figure 51.

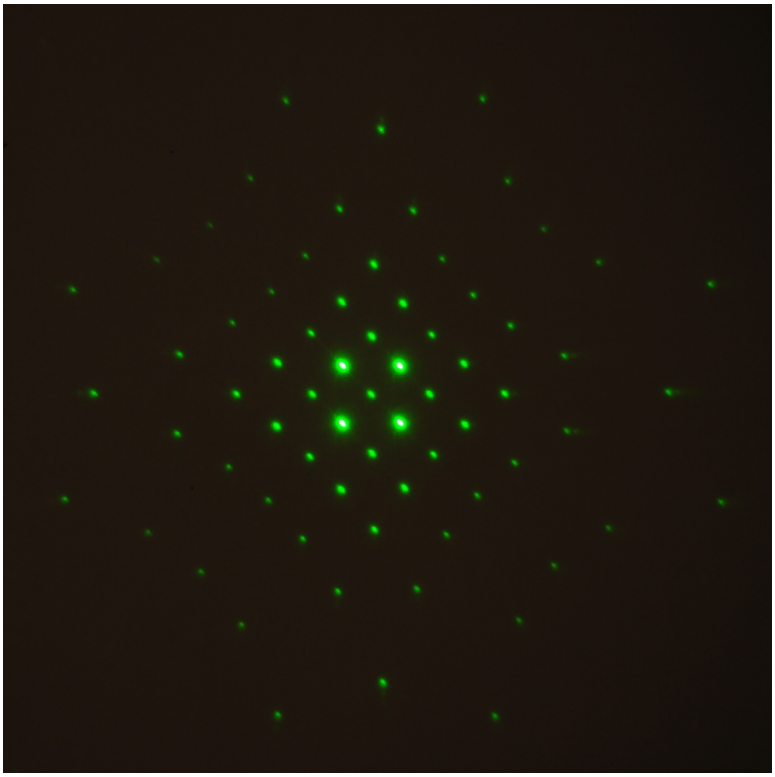

Figure 51: Diffraction pattern for test wafer

The diffraction pattern show the four bright second-order spots, as expected, but there are considerable zero, second and further even-order artefacts. These artefacts are a direct consequence of fabrication imperfections in terms of uniformity for the zero-

order and etch-depth for the other even orders. We currently do not have the instrumentation to measure the individual diffraction orders accurately. An attempt was made to measure the relative intensities of the orders from the photo but the pixels on the sensor tend to saturate quickly with the intensity levels shown; and the sensor does not have sufficient dynamic range for the measurement.

The 22 mm square design was also implemented and fabricated as a binary hologram. However, there was no image observed for a number of optical scenarios including spot illumination, collimated beam and use of focussing optics. The fabrication imperfections were observed to be similar in nature to those of the test wafer.

Note that the SMC is not able to align features down at one or two microns, so making a multi-level DOE is impractical.

4.2.4 DOE and SLM verification techniques

A number of techniques were developed alongside this work to gain confidence in the validity of the computer-generated holograms. The first method involved the SLM by running designs with features that are integer sub-multiples of the 19 μm SLM pixel pitch e.g. 9.5 μm (½), 4.75 μm (¼) or 2.375 μm (⅛). These were replayed on the SLM simply by subsampling the CGH phase array, for example, a 4.75 μm array can be subsampled by taking every fourth phase value. The *phase2im* program takes care of this automatically. It also truncates the array to the resolution of the SLM. The second method is a rigorous Fourier Optics-based theoretical framework that is implemented in a Matlab simulation environment, which is described in the following chapters.

4.3 Acknowledgements

I would like to acknowledge Dr Will Hossack, Edinburgh University School of Physics and Astronomy, for the use of the Holoeye SLM and the Edinburgh Parallel Computer Centre EPCC for the use of the Tesla Fermi GPU machine. I am also grateful to Dr Khaled Benkrid for the use of his GPU machine and for his advice on the best tools for GPU programming as well as optimisation hints on use of memory between the CPU host and GPU card. I would also like to thank Dr Chris Huntley and Dr Camilla Dunare at the SMC for fabricating the DOEs.

The CGH work was partially funded by a TTOM (Technology Transfer Opportunity Mechanism) Award from the Scottish Optoelectronics Association in collaboration with Edinburgh University. The CGH fabrication work was funded by a SUPA (Scottish Universities Physics Alliance) Award.

4.4 Conclusions and further work

The exercise proved to be quite technically challenging for the SMC as they are unaccustomed to running optical designs. In the end, a poor-quality binary phase test pattern was all that could be achieved and multi-level designs would need to be made by a fab that is more experienced in diffractive optics. A number of such fabs have been identified and are willing to collaborate with us: Silios (FR), Digital Optics (USA) and the Max Plank Institute (DE). Unfortunately, these are outside the UK and they are rather expensive. Further funding would be required to approach these foundries.

The source distributions described in this chapter are mostly one or two-dimensional. The direct convolution approach can handle 3D point data but the other models are constrained to 2D source and observation planes. This is because of the computational advantages gained via the 2D FFT vs direct convolution. Although it is possible to introduce 3D FFTs for 3D point data, the advantages are lost in terms of memory and computation time. This is because the source volumes tend to be rather sparse, it is often faster to use direct convolution for propagation modelling instead of Fourier approaches.

I believe that the toolchain I have developed to take a point cloud and generate photolithographic masks is a powerful one, and one that does not readily exist. This will be useful for any further photolithographic work. The phase mask tool currently only uses simple 1D polygon descriptions, this could be optimised further to 2D primitives and gain more compression. The *phase2im* tool could be extended to multi-level quantisation as it is currently constrained to binary.

The SLM verification technique is very powerful and useful in providing confidence with the DOE design before committing to fabrication, despite the shortcomings in implementation. The work going forward will focus on the SLM with theoretical and numerical approaches.

Chapter 5 - Fourier Optics Modelling

The foundation of scalar diffraction theory is described in Appendix C starting with Maxwell's equations with the key assumption that the light field can be represented as a

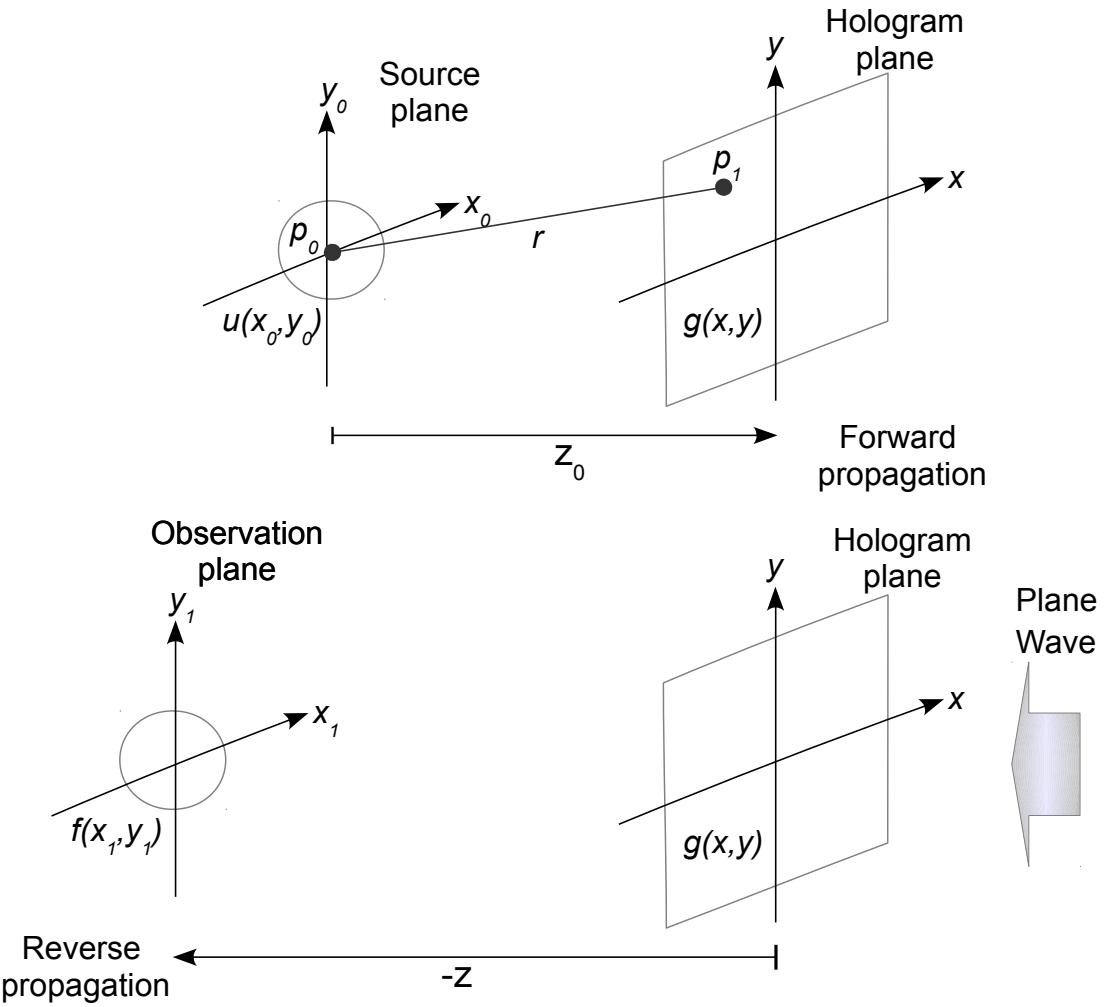

Figure 52: Fourier Optics recording (above) and reconstruction (below) models

complex scalar value. The analysis leads to Rayleigh-Sommerfield, Kirchoff, Fresnel and Fourier models of propagation via successive simplifications as a function of distance, *z*, from the hologram plane [95]. The theory is based on modelling the source image as a set of single-photon emitters propagating in free space and combining the scalar fields linearly in the hologram plane. This leads to the notion of the free-space impulse response that can be used to model propagation of an arbitrary source in a manner similar to a filter in electronics. The Fourier Transform can be applied to the situation to reduce the spatial convolution operation, between the source and free-space impulse response, to a simple multiplication in the spatial frequency domain. This frequency domain method is known as the angular spectrum approach, together with the associated overheads of forward and reverse transform operations.

The algorithms build on the DOE and SLM work from the previous chapter, based on the classical direct convolution method, which is computationally intensive but not very flexible. The work here is centred around a theoretical Fourier-Optics based hologram recording and reconstruction model, depicted in Figure 52. The Fourier optics geometry has three planes: a source plane with an arbitrary input shape *u(x,y)* that is propagated to the hologram plane, at a distance of z_0. The hologram plane *g(x,y)* is recorded (forward propagated) via the point source emitter model (upper part of Figure 52) that relates a single point $p_0(x_0,y_0)$ in u to a point $p_1(x_1,y_1)$ in *g* separated by a distance *r*, given by the relation:

$$r = \sqrt{(x-x_0)^2+(y-y_0)^2+z_0^2} \tag{30}$$

For reconstruction, or reverse propagation, the hologram plane is illuminated with a uniform monochromatic plane wave that is diffracted through the same Fourier-Optics propagation model to reconstruct the observation plane image *f(x,y)* at a distance of *-z* (lower part of Figure 52). Note that *-z* means that the illumination is behind the hologram plane.

The geometrical configuration of Figure 52 is illustrated mathematically in Figure 53 as a combined recording and reconstruction signal flow. This model is based on two levels of propagation: one for recording the hologram from the source at z_0 (forward propagation) and the other to reconstruct the observed image from the hologram at a distance *-z* (reverse propagation).

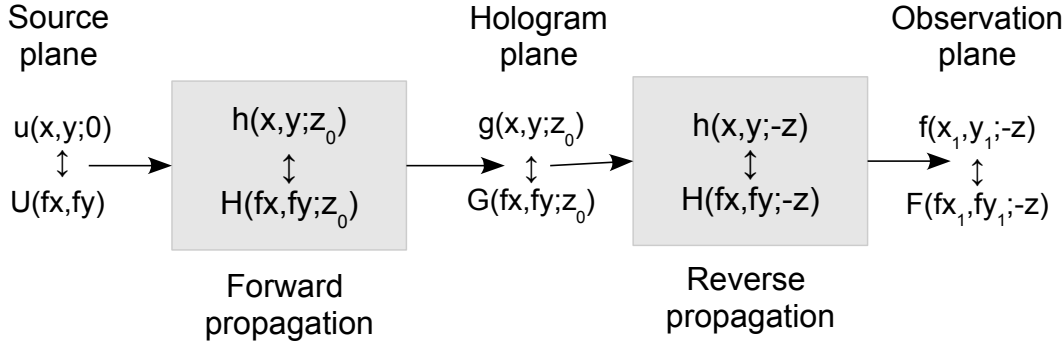

Figure 53: Combined forward and reverse propagation model

Full details of the mathematics behind these models are given in Appendix C. Figure 53 illustrates the functions in both spatial *(u,h,f)* and spatial frequency *(U,H,F)* domains which are related through the Fourier Transform. It is more computationally and mathematically expedient to show the combined propagation model in the spatial frequency domain.

The hologram recording, or forward propagation model, is given by:

$$G(f_x, f_y; z_0) = U(f_x, f_y; 0) H(f_x, f_y; z_0) \tag{31}$$

where f_x and f_y are spatial frequencies $H(f_x,f_y)$ is the angular spectrum of free space [58]. The reconstruction model applies the same propagation but in the opposite direction to yield the final reconstruction field, $F(f_x,f_y)$:

$$\begin{aligned} F(f_x, f_y; z) &= G(f_x, f_y; z_0) H(f_x, f_y; -z) \\ F(f_x, f_y; z) &= U(f_x, f_y; 0) H(f_x, f_y; z_0) H(f_x, f_y; -z) \end{aligned} \tag{32}$$

Noting that the Kirchoff and Fresnel forms of the angular spectrum expression are purely a function of phase i.e. $H(f_x,f_y;z)=exp(jkz\ \varphi(f_x,f_y))$ hence $H(f_x,f_y;-z)=H^{-1}(f_x,f_y;z)$, so:

$$F(f_x, f_y; z) = U(f_x, f_y; 0) \frac{H(f_x, f_y; z_0)}{H(f_x, f_y; z)} \tag{33}$$

In the special case where $z=-z_0$, then $F(f_x,f_y;z_0)=U(f_x,f_y;0)$, so the original source plane image function U is reconstructed. Note that the spatial domain reconstruction function is given by the inverse Fourier Transform of $F(f_x,f_y;z) \rightarrow f(x,y;z)$.

The continuous or analogue forms of the Fourier Transform are replaced by its digital equivalent via the Discrete Fourier Transform (DFT), or its numerically optimised version, the Fast Fourier Transform (FFT). The spatial dimensions and spatial frequencies are discretised, the minimum features thus have to obey the Shannon sampling theorem to avoid aliasing. The minimum features depend on the spectrum of the source image, such that the spectral sampling rate is at least twice the maximum source spatial frequency. This issue is discussed in the next section with band-limited input signals. The various transfer functions for the different models of propagation are shown in the table below:

Type	Impulse Response	Angular Spectrum
Kirchoff	$\frac{1}{j\lambda}\frac{z\exp(jkr)}{r^2}$	$\exp\left(-jkz\sqrt{1-\lambda^2(f_x^2+f_y^2)}\right)$
Fresnel	$\frac{\exp(jkz)}{j\lambda z}\exp\left(\frac{jk(x^2+y^2)}{2z}\right)$	$\exp(-jkz)\exp\left(j\lambda\pi z(f_x^2+f_y^2)\right)$
Fraunhoffer	1	1

Table 3: Fourier Optics propagation models

All of these propagators are implemented as Matlab functions (see section 5.3.3

appendix IV).

5.1 Propagation of a Gaussian source signal

The previous section together with Appendix C present an overview of various propagation and reconstruction models. These models can be applied to a source signal in order to gain an insight into the behaviour and interaction between the signal and the propagator. Consider a two-dimensional Gaussian signal *u(x,y)* propagated along the z-axis via Fresnel propagation *h(x,y:z)*:

$$u(x,y)=\exp\left(\frac{-\pi(x^2+y^2)}{2a^2}\right) \quad and \quad h(x,y;z)=\exp\frac{(jkz)}{j\lambda z}\exp\left(j\frac{k(x^2+y^2)}{2z}\right) \tag{34}$$

The hologram plane *g(x,y;z)* is then given by the convolution of these terms through the forward propagation model (Figure 53):

$$g(x,y;z) \;=\; u(x,y)*h(x,y;z) \tag{35}$$

This may be solved analytically via a variety of techniques including direct integration or Fourier-Transform methods to yield:

$$g(x,y;z)=\frac{2a^2}{2a^2-j\lambda z}\;\exp\left(\frac{\pi(4a^2z-j\lambda(x^2+y^2+2z^2))}{\lambda^2 z+j2a^2\lambda}\right) \tag{36}$$

This is an exact analytic expression encapsulating the hologram of a Gaussian voxel of dimension *a* at a distance *z*. The field corresponding to this hologram of a Gaussian voxel, located at z_0, is reconstructed via the reverse propagation model (Figure 53):

$$f(x,y;z)=g(x,y;z_0)*h(x,y;-z) \tag{37}$$

Again, this can be solved analytically in the same manner to yield:

$$f(x,y;z)=\frac{2a^2}{2a^2-j\lambda(z-z_0)}\;\exp\left(\frac{\pi(4a^2(z-z_0)-j\lambda(x^2+y^2+2(z-z_0)^2))}{\lambda^2(z-z_0)+j2a^2\lambda}\right) \tag{38}$$

It is apparent that when $z=z_0$, the original signal can be recovered:

$$f(x,y;z_0) \;=\; \exp\left(\frac{-\pi(x^2+y^2)}{2a^2}\right) \;=\; u(x,y) \tag{39}$$

It can be seen that the phase of the signal is modulated by the Fresnel chirp function. Indeed, this is the only factor that determines the position of the voxel at z.

5.2 Source signal bandwidth

The Fourier Optics theory presented in Appendix C assumes continuous or analogue signals. However, once we apply digital techniques, certain constraints have to be placed on the signals being sampled. This section examines sampling in more detail in order to gain an insight into the requirements for hologram plane representation.

The approach requires consideration of the source signal in both spatial and spatial frequency domains [96, p. 14]. Fourier Optics modelling usually deals with two dimensional signals $u(x,y)$. In order to simplify the analysis, assume that the signal is separable in x and y, i.e. $u(x,y)=u_x(x)u_y(y)$. Consider a general spatial signal, $u(x)$, which is symmetric, continuous and non-periodic; and its frequency domain representation $U(f)$, computed via the Fourier Transform. Define the Total Spectral Energy, E_T in terms of intensity as:

$$E_T = \int_{-\infty}^{+\infty} |u(x)|^2 \, dx \tag{40}$$

From Parceval's theorem, this is equivalent to its spatial frequency representation:

$$E_T = \int_{-\infty}^{+\infty} |u(x)|^2 \, dx = \int_{-\infty}^{+\infty} |U(f)|^2 \, df \quad and \quad U(f) = \int_{-\infty}^{+\infty} u(x) \exp(j2\pi fx) \, dx \tag{41}$$

Where $U(f)$ is calculated via the continuous Fourier Transform, which can be tricky. Often, the transform can be derived analytically or found in mathematical tables. E_T, however, is usually infinite in extent and thus needs to be constrained or band-limited in order to make it finite. Now, define a bandwidth constrained version of this function, E_B, over the symmetric frequency range $-B$ to $+B$ such that it contains 98% of the Total Spectral Energy, E_T :

$$E_B = \int_{-B}^{+B} |U(f)|^2 \, dx = 0.98 \, E_T \tag{42}$$

Next, calculate the effective bandwidth, B, to satisfy the constraint, which is usually computed numerically. The corresponding sampling interval is then given by translating

the double-sided spectrum (effective bandwidth) back into the spatial domain:

$$\Delta x = \frac{1}{2\mathrm{B}} \tag{43}$$

This is a statement of the Nyquist-Shannon sampling criterion [97,98], where the minimum sampling rate is twice the maximum spatial frequency for signal reconstruction. If *g(x)* is finite over the interval A then the total number of samples required to represent the signal is given by:

$$N_s = \frac{A}{\Delta x} \tag{44}$$

The choice of 98% is purely arbitrary but it needs to encapsulate a reasonable amount of the signal [96] across many different types of signals e.g. Gaussian, rectangular, circular etc, and is readily extended to two dimensions provided the source signal is separable. To summarise the steps:

1. Compute the Total Spectral Energy, E_T using $u(x)^2$ in eq. (40)
2. Derive the Fourier Transform *U(f)*
3. Compute the Effective Bandwidth *B* using $E_B=0.98E_T$, using $U(f)^2$ eq (42)
4. Compute the sampling interval from *B*, using eq (43)

5.2.1 Gaussian signal

The process can be illustrated by means of an example. Consider a real-valued Gaussian source in one dimension:

$$u(x) = \exp\left(-\frac{\pi}{2a^2}x^2\right) \tag{45}$$

Where *a* is constant and defines the width of the input source. The Total Spectral Energy is given by the improper integral in the spatial domain (Step 1):

$$E_T = \int_{-\infty}^{+\infty} u(x)^2 dx = \int_{-\infty}^{+\infty} \exp\left(-\frac{\pi}{a^2}x^2\right) dx \;=\; a \tag{46}$$

Note the modulus is not written as this is a real-valued signal. The Fourier transform of a Gaussian function is also a Gaussian (Step 2):

$$U(f)=\int_{-\infty}^{+\infty} u(x)\exp(-j\,2\pi\,x\,f)\,dx=\sqrt{2}\,a\exp\left(-2\pi a^2 f^2\right) \tag{47}$$

The solution of the improper integral $U(f)^2$ yields the same result as eq. (46), also real-valued, as expected:

$$E_T=\int_{-\infty}^{+\infty} U(f)^2\,df=\int_{-\infty}^{+\infty} 2a^2\exp\left(-4\pi a^2 f^2\right)df \;=\; a \tag{48}$$

which is also proof of Parceval's theorem. However, we are interested in the definite integral of equation (48) across the constrained bandwidth $\pm B$ such that:

$$E_B=\int_{-B}^{+B} 2a^2\exp\left(-4\pi a^2 f^2\right)df=0.98\,E_T \tag{49}$$

Next solve for B (Step 3):

$$\int_{-B}^{+B} a\exp\left(-2\pi a^2 f^2\right)df=a\,erf\left(2\,a\,B\sqrt{\pi}\right)=0.98\,a \tag{50}$$

Whereby *erf(y)* is the error function defined as:

$$erf(\psi)=\frac{2}{\pi}\int_0^{\infty}\exp(-\psi^2)\,d\psi \tag{51}$$

Equation (50) can be rearranged to yield B:

$$B=\frac{erf^{-1}(0.98)}{2\,a\sqrt{\pi}}=\frac{0.4640}{a} \tag{52}$$

Where erf^{-1} is the inverse error function, whose argument cannot exceed ±1. This simple expression is a direct relation between the size of a Gaussian signal source and the bandwidth required to satisfy the 98% constraint. If we choose a Gaussian voxel with a value of one millimetre for a, we get:

$$B=464.0\,m^{-1} \tag{53}$$

B is expressed in units of m^{-1}, for spatial frequency. As a quick check on this result, consider the $U(f)^2$ in terms of a statistical normal distribution of the type:

$$f(\alpha)=\exp\left(\frac{-\alpha^2}{2\sigma^2}\right) \tag{54}$$

Where σ represents one standard deviation, so comparison with equation (46), yields:

$$\sigma=\frac{1}{a\sqrt{8\pi}}\approx 200.0\,m^{-1}\ for\ a=1\,mm \tag{55}$$

As a statistical rule of thumb, widths of ±2σ and ±3σ encapsulate 95% and 99.7% of the signal, hence $B_{2\sigma}$ = 400.0 m^{-1} and $B_{3\sigma}$ = 600.0 m^{-1}, so B=464.0 m^{-1} for 98% appears to be reasonable.

Finally, translate back into the spatial domain by considering the full double-sided spectrum (Step 4):

$$\Delta x=\frac{1}{2B}\ \approx\ 0.171\,mm \tag{56}$$

Hence the Gaussian signal *g(x)* requires a sampling interval of 0.171 mm in order to reconstruct it within a spectral bandwidth of 98%. In practice, sampling would be at a much finer resolution in order to attain sufficient diffraction. Nevertheless, this is a good starting point to gain insights into the bandwidth requirements of voxel sampling.

5.2.2 Top-hat signal

The top-hat or box is another common signal found in optics, and we repeat the same steps to obtain the bandwidth required for such a function, which has a value of unity in the interval *-0.5a< x < 0.5a.* The Total Spectral Energy is therefore simply:

$$E_T=\int_{-\infty}^{+\infty}u(x)^2dx=\int_{-0.5a}^{+0.5a}1^2dx\ =\ a \tag{57}$$

Which contains the same amount of energy as the previous signal. The corresponding frequency domain function is given by the *sinc* function:

$$U(f) = a\,sinc(a\pi f) \quad where \quad sinc(\alpha) = \frac{\sin(\alpha)}{\alpha} \tag{58}$$

The Total Spectral Energy is also given by the improper integral which is consistent with eq. (58) and Parceval's theorem:

$$E_T = \int_{-\infty}^{+\infty} U(f)^2 df = \int_{-\infty}^{+\infty} a^2\,sinc^2(a\pi f) = a \tag{59}$$

Now apply the band-limiting constraints:

$$E_B = \int_{-B}^{+B} a^2\,sinc^2(a\pi f) = 0.98\,a = \frac{2\pi a B\,Si(2\pi a B) + \cos(2\pi a B) - 1}{\pi^2 B} \tag{60}$$

Where *Si* is the Sine Integral function defined as:

$$Si(\chi) = \int_0^{\chi} \frac{\sin(\beta)}{\beta} d\beta \tag{61}$$

Eq. (60) is not easy to solve or express analytically in terms of B, so the best approach is to solve this numerically to yield:

$$B \approx 5225.0 = \frac{5.22}{a} \tag{62}$$

The spatial resolution is then:

$$\Delta x = \frac{1}{2B} \approx 0.09569\,mm \tag{63}$$

Note that the spectral requirements for the top-hat function are over ten times that of the Gaussian. This is because of the ringing in the frequency domain caused by sharp discontinuities on the edges of the top-hat profile.

5.3 Matlab algorithms

The theoretical models built up in this chapter are implemented as a series of algorithms in Matlab for the purposes of simulation and design exploration. We start by discretising the signal through sampling, followed by a review of important functions or

algorithms from the literature, including: quantisation, propagation, optimisation and other effects. A broad list of Matlab function headers are presented in Appendix D. These algorithms can work stand-alone or in cascade as signal processing blocks. The algorithms form the basic building blocks of a Fourier Optical modelling framework that culminates in a fully-blown design tool presented in the next chapter.

5.3.1 Sampling theorem in 2D

The previous sections provided some insights into the bandwidth requirements for various types of signals. These can be used as a basis for signal sampling by expanding in two dimensions. The Nyquist-Shannon sampling theorem states that a bandlimited signal over a finite region may be represented exactly if the sampling rate is at least twice the highest frequency within the signal [97,98]. This was initially proposed for one dimensional signals in communication systems, involving sampling the signal by a series of impulse functions or a comb function. This may be extended to two dimensions via a matrix of impulses or a 'bed of nails' function as shown in Figure 54 where X and Y are the intervals between successive samples. This co-ordinate system is represented in Matlab as set of Meshgrids.

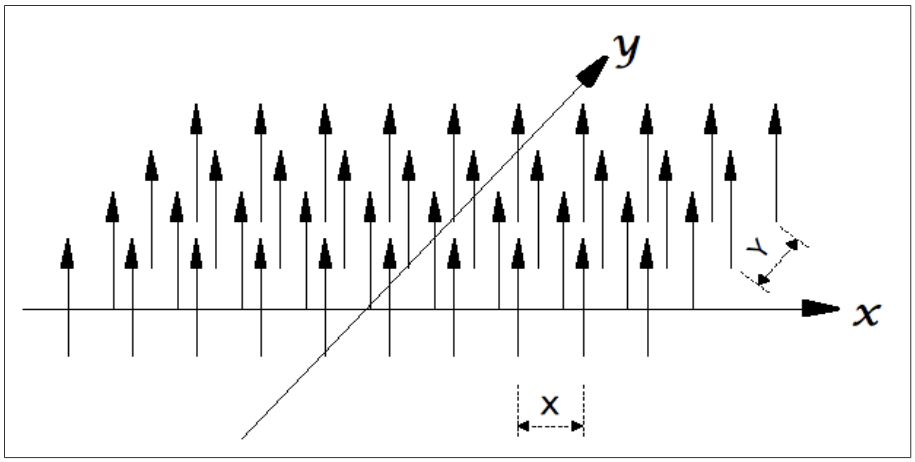

Figure 54: Sampling function in 2D

A signal or field, $g_s(x,y)$ thus sampled can be written as:

$$g_s(x,y) = XY \sum_{m=1}^{M} \sum_{n=1}^{N} g(nX, mY) \tag{64}$$

The spectrum of g_s is given by the Discrete Fourier Transform (DFT) G_s:

$$G_s(u,v)=\frac{1}{\sqrt{MN}}\sum_{m=1}^{M}\sum_{n=1}^{N}g(n,m)\exp\left[-2\mathrm{j}\pi\left(\frac{um}{N}+\frac{vn}{M}\right)\right] \quad (65)$$
$$where \quad 1\le u\le M \ \ and \ 1\le v\le N$$

Where u and v are spatial frequencies. An optimised version of the DFT (9) is the Fast Fourier Transform or FFT algorithm where the number of sampling points is a power of 2. This leads to a computationally efficient generation of samples. In general, G_s is a (M,N) matrix of discrete complex values. Since the sampling function is periodic, the corresponding frequency spectrum of the sampled signal is also periodic. The sampling rate and number of samples must satisfy the Shannon-Nyquist criterion to prevent aliasing which means sampling at least twice the signal bandwidth along both axes:

$$X\ge\frac{1}{2\mathrm{B}_X} \quad and \quad Y\ge\frac{1}{2\mathrm{B}_Y} \quad (66)$$

Where B_X and B_Y correspond to the bandwidth criteria identified in section 5.2. If these conditions are met then the signal can be reconstructed exactly through low pass filtering. The ideal low pass filter is given by the 2D rectangular function over these bandwidths applied in the spatial frequency domain:

$$H(u,v)=rect\left(\frac{u}{2\mathrm{B}_X}\right)rect\left(\frac{v}{2\mathrm{B}_Y}\right) \quad (67)$$

The filtered signal is thus represented in both spatial and frequency domains through the following relations:

$$G=G_s(u,v)H(u,v) \quad \Leftrightarrow \quad g(x,y)=g_s(x,y)*h(x,y) \quad (68)$$

Where $h(x,y)$ is the impulse response of $H(u,v)$ and * is the convolution operator applied in the spatial domain. These are linked through the Fourier Transform. Hence, $g(x,y)$ may be reconstructed exactly, noting that the ideal low pass filter Fourier transforms into a sinc function:

$$g(x,y)=\sum_{m=-\infty}^{\infty}\sum_{n=-\infty}^{\infty}g\left(\frac{n}{2\mathrm{B}_X},\frac{m}{2\mathrm{B}_Y}\right)sinc\left[2\mathrm{B}_X\left(x-\frac{n}{2\mathrm{B}_X}\right)\right]sinc\left[2\mathrm{B}_Y\left(y-\frac{n}{2\mathrm{B}_Y}\right)\right] \quad (69)$$

Which enables the original signal to be recovered.

5.3.2 Quantisation: amplitude and phase

Although the source signals in Fourier Optics are usually real-valued, the signals and computed hologram plane generally have both amplitude and phase characteristics as a result of the Fourier Optical and other processing. In order to represent these signals in digital form, it is necessary to perform quantisation on both amplitude and phase components of the hologram. Most LCD-based amplitude-modulated hardware are designed for 8-bit or 256 levels, which basically peg the amplitude quantisation. It has been seen in the previous chapter that SLMs also have 256 phase levels as they are based on LCD hardware. However, 256 levels for phase quantisation is overkill, as we saw for DOEs, where only eight levels are required to attain 95% diffraction efficiency; whereas 16 levels reach 98% efficiency with diminishing gains for more levels [57, p. 82].

The Matlab functions to perform quantisation are: *Qampl.m* and *Qphase.m*.

5.3.3 Propagation functions

The various propagation models are described in detail at the start of this chapter, including Kirchoff and Fresnel diffraction for both impulse response and angular spectrum approaches using the Fourier Transform. The propagation functions are variously implemented as: *HrsIR.m* (Kirchoff Impulse Response), *HfrIR.m* (Fresnel Impulse Response), *HrsAS.m* (Kirchoff Angular Spectrum) and *HfrAS.m* (Frenel Angular Spectrum).

The functions *forwardProp.m* and *reverseProp.m* respectively perform forward and reverse propagation.

The direct convolution function is very similar to the earlier CGH.m series of functions (section 4.1), now implemented as: *ConvProp2.m*

5.3.4 Bipolar Intensity algorithm

The bipolar intensity algorithm is a means of computing amplitude-only diffraction patterns proposed by Lucente [34]. This is a real-valued pattern derived from interfering an off-axis plane wave with set of object points via Fresnel propagation using direct convolution based on equation (5). Lucente's optimised version is constrained to horizontal parallax to reduce the computations [99]. The implemented version of this algorithm has an on-axis plane wave and uses the standard propagators but takes the real value and discards the complex part of the computation.

5.3.5 Phase randomisation

The source signals in sections 5.2.1 and 5.2.2 are real-valued without a phase

component. This implies that the signals are perfectly smooth and without any surface artefacts, something which is not found in practice where objects are rough relative to the wavelength of light. This roughness or random variation of surface profile can give rise to substantial amplitude variations across the diffracted field. Specifically, this phenomenon is characterised as phase variations across the object, which is observed as speckle when an object is illuminated with coherent radiation. Such surfaces can be modelled by introducing a random phase multiplier:

$$s(x) = u(x)\exp\left(j\,2\pi\,n(\varphi)\right) \tag{70}$$

Where *u(x)* is the smooth source signal and the random term, *n(φ)*, varies between 0 and 1, so the expression *exp(j2π n(φ))* represents random phase values between 0 and 2π. Note that the magnitude of the source signal remains intact. The degree of roughness or light diffusivity depends on the statistics of the random variable *n(φ)*. A uniformly distributed *n(φ)*, where each point is uncorrelated with its neighbours, represents a very rough or highly diffuse surface. This introduces high frequency spectral components into the source signal that can lead to aliasing effects. The power spectrum can be smoothed by using non-uniformly distributed phase variations or phase values that do not vary significantly compared to neighbours [100]. This can be performed by a number of image processing algorithms that can generate a uniformly distributed set of values and smooth them using filtering through convolution with an appropriate kernel (median, Gaussian, mean) [101].

The initial version of the Matlab algorithm used a uniformly distributed noise source across the entire image. This was superseded by a version that applied the uniform source to just one quadrant of the image, which was subsequently mirrored and reflected across the other quadrants. This has the effect of phase-conjugation across the image, resulting in a symmetric representation in the Fourier domain.

The random phase Matlab algorithms are: *randphase.m* for the pure version and *randphase2.m* for the conjugate version.

5.3.6 Amplitude randomisation

This is similar to the phase randomisation algorithm except that the noise *n(x)* is applied to the source amplitude to yield a varying randomised absorption across the source signal, equation (71). The distribution of the noise source (linear, Gaussian) determines surface profile properties, depending on statistical parameters (mean, variance). This can be used to change the appearance of the source signal and combat

issues around speckle.

$$s(x)=u(x)n(x) \tag{71}$$

The algorithm is implemented in *randampl.m*

5.3.7 Iterative Fourier-Transform algorithm

Section 5.3.5 noted how arbitrary phase can be assigned to amplitude-only signals in order to model surface profiling in the hologram plane. The hologram plane now contains both amplitude and phase characteristics. This can be taken a stage further by representing amplitude-only signals as pure phase holograms. Phase-only computer generated holograms are computed using Iterative Fourier Transform techniques, introduced in section 1.4.2 starting with the Gerchberg-Saxton (G-S) algorithm [65], which is an error-reduction algorithm aimed at optimal phase retrieval between an input plane and an output plane via the intermediate hologram plane. The input and output planes are related via the Fourier Transform. The subject has been studied extensively with reviews and comparisons [102] of the various improvements and enhancements to the basic G-S algorithm [103,104]. IFT algorithms have been designed and optimised for the following applications:

- Spot generation or beam-splitting
- Beam-shaping or profiling e.g. Gaussian laser to a flat-top
- Image pattern generation

These applications have different characteristics, for example, the first one is designed to harmonise energy distribution across the spots, whereas the second is concerned with harmonising intensity and the last one deals with image fidelity. As a consequence, the choice of algorithm is driven by a merit function that can optimise power distribution, error (between desired and computed signals) or the signal-to-noise ratio. Many IFT algorithms deal with far-field Fourier holograms, however, we are interested in near-field Fresnel or Kirchoff holograms, which apply these propagators between the input/output and hologram planes.

The algorithm starts with a random phase distribution for the input plane (from the previous section) and applies a forward propagator (e.g. Fresnel) to compute the hologram plane. The constraints are then applied to the hologram plane where the amplitude is set to unity but the phase is preserved. This constrained hologram is then

reverse propagated to the output plane where its intensity is compared with the original input to generate an error merit function. The input plane phase is then adjusted and the algorithm repeats for a set interval. The algorithm is not guaranteed to converge on a solution. The goal of the iterative algorithm is to minimise the merit function and the algorithm is believed to converge if it satisfies one or more of the following criteria:

- a minimum target merit function is met
- the merit function no longer changes significantly
- a maximum number of iterations or a timeout is reached

The implemented Matlab program applies all of these criteria. The Iterative Fourier Transform Matlab algorithm is called *ITFT.m*; a version that takes the propagator into account is: *ITFT2.m*

5.3.8 Impulse voxel expansion (X-Y plane)

The impulse expansion algorithm takes any single voxel e.g. a Gaussian shape and replicates it across the source plane. This is achieved by convolving the voxel with a series of impulses arranged across the source plane. The convolution operation is simplified in the frequency domain which reduces to a multiplication of the Fourier transformations of these signals. This approach has the advantage that the voxel only needs to be computed once and generation of the impulses is relatively straightforward.

The impulse expansion algorithms is called: *impulseGrid.m*

5.4 Conclusion

This chapter presented the building blocks of a framework for modelling and algorithmic exploration of computer generated holograms. This includes both hologram pattern synthesis and reconstruction. Issues such as sampling common signals and signal bandwidth are explored and taken into consideration. Finally, a review of key algorithms from the literature are presented. Although most of these are standard functions, I have not found anything in the literature that brings them together in one place. These were implemented as individual functions in Matlab and the library is presented in Appendix D. The ideas presented here are taken forward into the next chapter to develop an integrated simulation and synthesis tool around digital Fourier Optical analysis.

Chapter 6 - FOURIOT: Fourier Optics Design Tool

The Fourier-Optics mathematical model for recording and reconstruction is implemented as framework of numerical algorithms written in Matlab (7.12.0 r2011a). As the amount of code and algorithms proliferated, it became increasingly difficult to manage all of the combinations of functionality that were possible. It also became difficult to remember the sequence of tasks along the signal processing chain, especially when it comes to researching combinations of scenarios. The approach was too ad-hoc, so I decided to wrap the image and signal processing functions into a general tool for Fourier Optical propagation and reconstruction. The tool, called FOURIOT, for FOURier-Optics Tool, is based on the Matlab Graphical User Interface (GUI) framework, depicted in Figure 55.

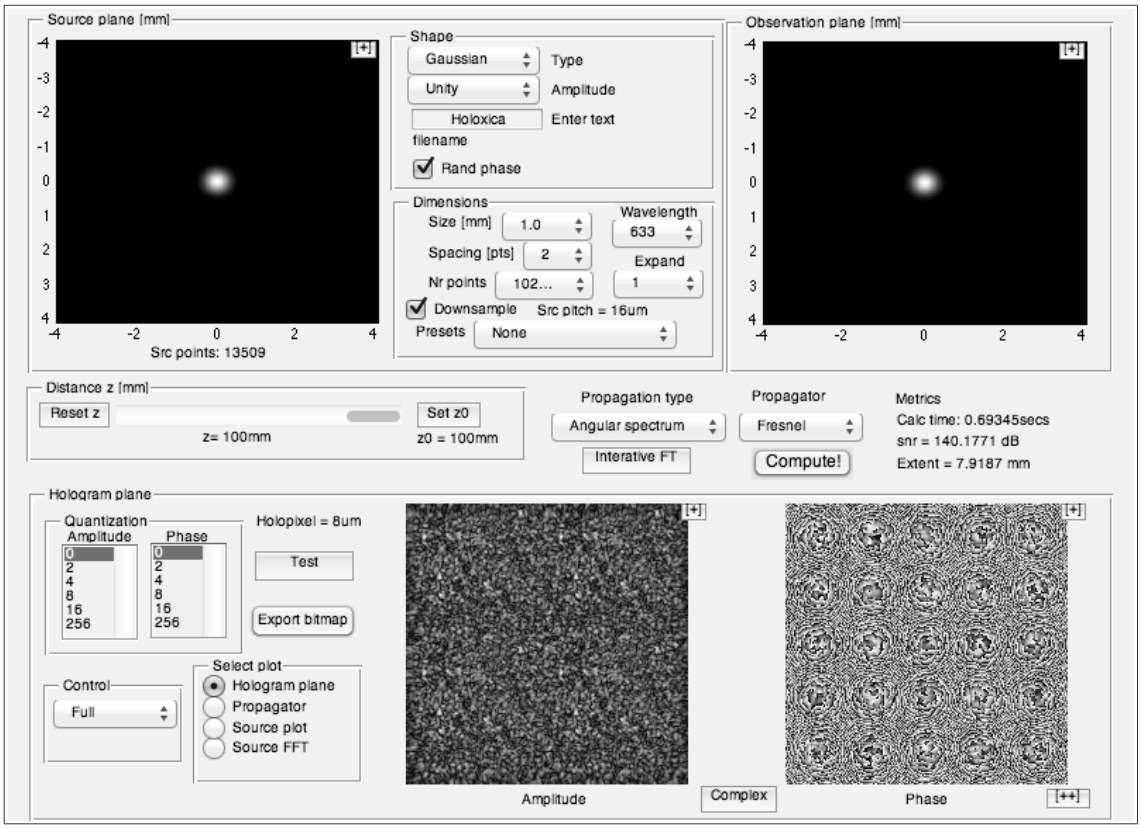

Figure 55: FOURIOT User Interface

Most optics tools tend to favour geometric or ray optics (e.g. Code V or Zeemax) and they do not generally handle diffractive optics very well. There are a few diffractive optics tools around but these are either expensive or aimed at laser design (beamshaping, cavities), simple fan-outs or optical elements. Example of such tools include a free Matlab-based library from Cambridge University and a recent commercial tool, VituaLab product by LightTrans. FOURIOT is specifically aimed at

image reconstruction and voxel exploration for holographic display research and design.

6.1 FOURIOT User Interface

The FOURIOT user interface has four zones that follow the hologram recording and propagation models from Figure 52 in section 5.

- Top left: source plane
- Middle: propagation parameters
- Bottom: complex hologram plane
- Top right: observation plane

FOURIOT works in units of millimetres between all planes. The source panel (top left, Figure 55) contains the source voxel or image to be propagated to the hologram plane (bottom panel) some distance (*z0*) away, determined by the propagation parameters (middle panel). Finally, the resulting image is depicted in the observation plane (top right).

An overview of the various planes and panels are presented in the following sections. There are many options available under the various menus, drop-down lists and buttons, so this description is not meant to be exhaustive and only the main features are presented together with some use-case scenarios.

6.1.1 Source plane (top-left)

The source plane comprises three panels: a graphic of the source intensity, the shape of the source image and dimensions or parameters to control some of the source images. The graphic of the source intensity has the axes in mm. The [+] icon on the top-right of the graphic expands the image to reveal further detail. The Nr points indicator shows the amount of non-zero points in the image, to get an idea of its complexity; an important parameter for some propagation algorithms.

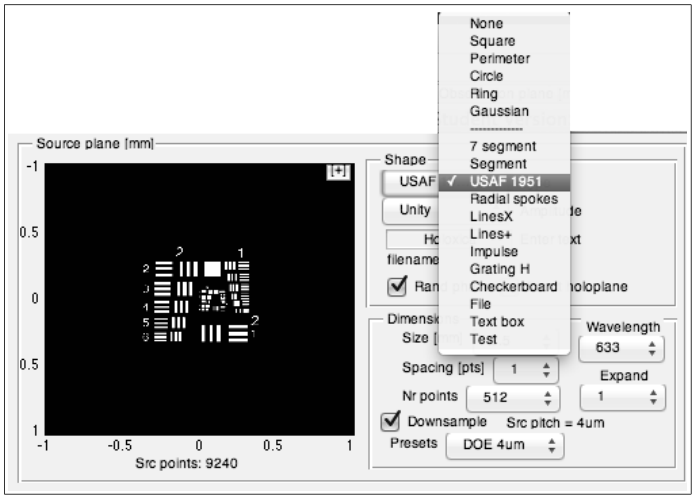

Figure 56: Source shape choices

It can be seen from Figure 55 that the default source image is a Gaussian shape. The source shape (Figure 56) can be selected from a set of shape primitives

(Gaussian, square, circular, perimeter, ring), segmented digit, preconfigured images (test patterns), gratings and other basic primitives (lines, impulse etc). Standard test patterns include the USAF 1951 resolution test chart (Figure 56) and the radial spokes target which gives a good visual impression of spatial resolution. In addition, user-defined plain text can be shown or even external bitmap images loaded from a graphics file (.bmp, .jpg etc). The amplitude option enables some randomisation of the amplitude characteristics of the source image (gaussian or pure random with or without an offset); the default is unity amplitude (no randomisation). The phase can similarly be randomised via the RandPhase option, set by default.

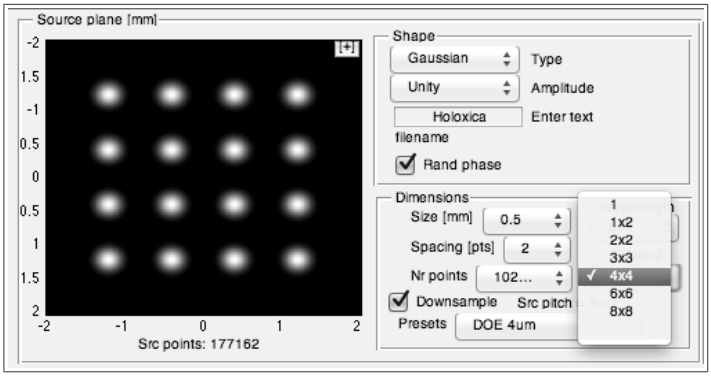

Figure 57: Source signal expansion

The Dimensions panel enables key parameters around the dimensions of the source plane to be set, and by implication, sets the dimensions for the other planes. This includes the coherent source illumination wavelength (default 633 nm). The size option sets the size (in mm) for Gaussian, square, circle, perimeter and seven segment. The Nr Points parameter sets the geometry of the source, hologram and observation planes; the values range from 512x512 to 3000x3000; the default is 1024x1024. The spacing between adjacent points in the source image can be selected with values ranging from 1 to 32 points; the default is 2 points. This parameter can be used to exercise a finer control on the spacing of the image points. For practical purposes, the spacing between adjacent points can be around 50 μm and the human eye would not notice. This can reduce the number of points in the image, which can be important for propagation algorithms based on direct convolution methods. This parameter effectively downsamples the source image and the *Downsample* checkbox (enabled by default) ensures that the source and observation plane images do not appear with too many gaps in the small graphic

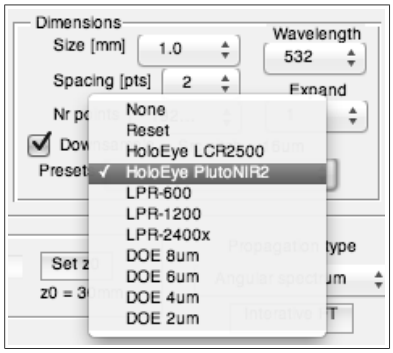

Figure 58: Preset templates

areas within the tool. The actual source spacing (in microns) is shown in the corresponding text box.

The *Expansion* option under *Dimensions* takes a single source image and replicates it using the impulse expansion algorithm (section 5.3.8). The default value is 1 (no expansion) and the expansion options range from 1x2, 2x2 to 8x8. Figure 57 Shows a Gaussian shape expanded are placed evenly across the source space on a 4x4 grid.

The *Presets* option under *Dimensions* (Figure 58) selects between a number of templates aimed at specific physical implementations including SLMs (Holoeye LCR-2500 and Pluto-NIR2), diffractive optical elements (phase-only) of varying dimensions (from 2 to 8 μm cell pitch) or high resolution laser printers (amplitude-only) between 600 to 2400 DPI. The source is depicted as an intensity (amplitude squared). The characteristics of each template is known, so, for example, the Holoeye LCR-2500 preset selects 1024x768 pixel source plane, 19 μm pixel pitch, 256 level phase-only quantisation etc.

6.1.2 Propagation panel (middle)

The propagation panel (across the middle, Figure 55) defines the distance between the source and hologram plane, *z0*, as well as the reconstruction distance, z, in the observation plane. Moving the slider on the panel, changes z, whereas the 'Set z0' button sets the value of *z0* to the current value of z, thus defining the location of the hologram plane. Thus, *z=z0* when the button is pressed. The value of *z0* is shown just above this button. The scale of the slider depends on the preset template (section 6.1.1) which is either 50-550 mm for a holopixel pitch of <20 μm or 100-1100 mm for holopixels >20μm. The default values of *z/z0* are 100 mm and 500 mm, which can also be selected via the 'Reset' button. For most templates, the propagator will be used in the 100-1100 mm regime, in which case the fine movement is 5 mm via clicking the arrow keys on slider and the coarser movement is in steps of 50 mm by clicking else where on the slider.

The user can choose the propagation model between Kirchoff, Fresnel (default) or Fraunhoffer (see Appendix C and section 5). Implementation options include: impulse response, angular spectrum (default), direct convolution or bipolar intensity. Warning – the direct convolution method may take a long time to run, depending on the size of the hologram (Nr Points) and the amount of information in the source image (Src Points).

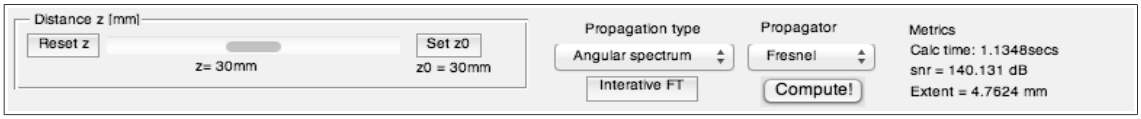

Figure 59: Propagation panel

The 'Iterative FT' toggle button applies the Iterative Fourier Transform for an improved calculation of phase-only profiles (section 5.3.7). Warning – this option can take a long time to reach a solution, although the maximum number of iterations is constrained to fifty. Progress is shown in the main Matlab window. This option is only available for impulse response and angular spectrum propagation models.

Note that convolution methods do not have an adequate reverse propagation model, so the angular spectrum method is used to approximate these cases.

The compute button repeats a calculation, if required. The calculation is a complete computation of the forward propagation model to compute the hologram plane, followed by the reconstruction model for the source plane. Use of this button is not necessary for normal operation since the calculation is automatically triggered by whatever action is performed. However, it may occasionally be necessary to, for example, trigger a random phase change (if selected).

Some basic metrics are included on the right of the panel, such as computation time and the signal-to-noise ratio (SNR) of the observed image with respect to the source. Note that the SNR is normalised to take into account differences in RMS energy between the source and observed images (Appendix D). The extent of the source is the diffraction-limited reach of each holopixel; a factor dependent on the wavelength, holopixel size and z0, the distance between source and hologram planes.

6.1.3 Hologram plane (bottom)

The hologram plane is the largest section of the GUI, which controls the representation of the hologram plane computed according to the propagation

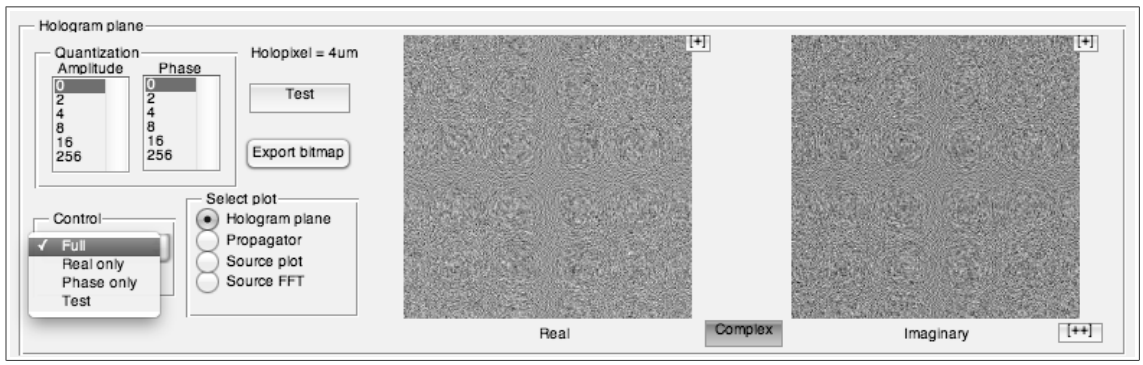

Figure 60: Hologram plane (real & imaginary views)

parameters. The main part of the pane depicts the complex diffraction pattern in amplitude/phase format (default) where the '*Complex*' button switches the pair of graphic views between this or real/imaginary representations as depicted in Figure 60. The amplitude profile is a normalised greyscale and the phase profile is represented by a greyscale value where black=0, π=halftone grey and white=2π phase shift. The '*select plot*' sub-panel can be used to select what is shown in the complex plots, the default being the hologram plane; where the user can select between the propagator (Kirchoff, Fresnel etc), the source plot (full complex version of Figure 56) or a 2D FFT of the source plot.

The '*Control*' sub-panel allows the selection of phase or amplitude-only transmission of the hologram. These are not set by default, so the hologram is fully complex. The user can choose between amplitude or phase-only implementations.

Quantisation of the amplitude or phase is performed via the '*Quantisation*' sub-panel. A value of zero (default) indicates no quantisation. The quantisation levels for amplitude or phase can be set independently (up to 256 levels). This can be used for implementation on e.g. an SLM, DOE or amplitude mask, which can be exported as bitmaps ('*Export bitmap*' button) for utilisation by other tools. The exporting button will only work if the phase or amplitude profile has been quantised, so it will warn you if this is required. If quantised, a file save dialogue box will open, allowing you to specify

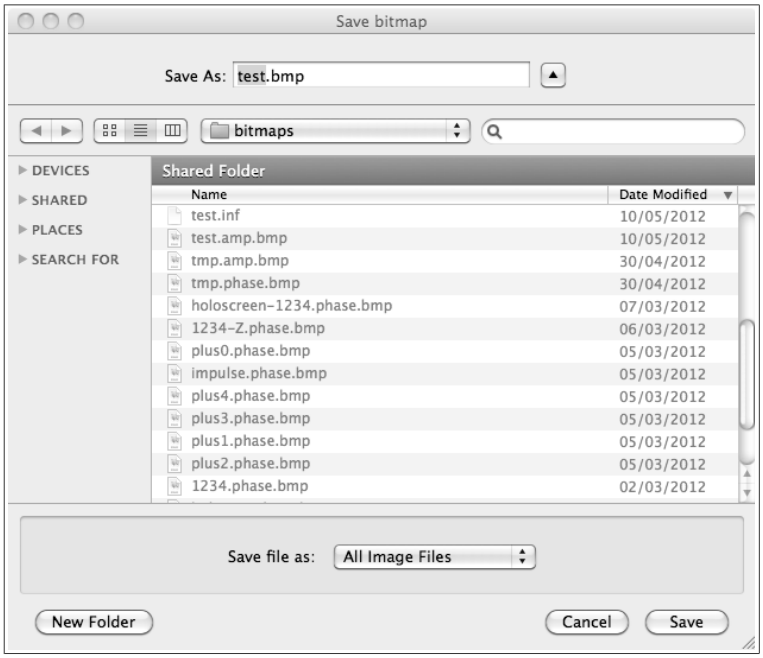

Figure 61: Exporting bitmaps

a filename or navigate to a directory to save the images. The default filename is *'test.bmp'* and the bitmaps are saved in the *.bmp* format; FOURIOT uses a default

'bitmaps' directory to save the files, Figure 61. FOURIOT knows whether either or none of these have been selected for the *'Export bitmap'* function and it appends *'.ampl'* and/or *'.phase'* to the export filename. The '*Quantization*' and '*Control*' sub-panel options are automatically configured by any preset template (section 6.1.1).

The export function will also save a *.inf* text file that contains some extra information about the image. This includes parameters such as the dimensions, propagation distance (z0), pixel pitch, wavelength and the preset. This information may be useful for later stage processing. An example of the text for the SLM preset is shown below:

```
pointsX=1024
pointsY=768
z0=5.000000e-01
pitch=1.900000e-05
lambda=6.328000e-07
preset=SLM
```

The '*Test*' button is used for testing new functions before they are actually integrated within the tool.

The [+] icons on the top-right of each plot expands the image to reveal further detail. The [++] icon on the bottom right hand corner shows a composition of all four plots (source, hologram amplitude/phase and observation planes) for documentation purposes (see section 6.2). Alternate invocations of these expansion functions depict the images with inversion on or off, which is convenient for saving the images for documentation purposes. For example, inverted images (i.e. black on white background) is preferable for presentation in print and is used in some of the figures presented in this chapter, for example.

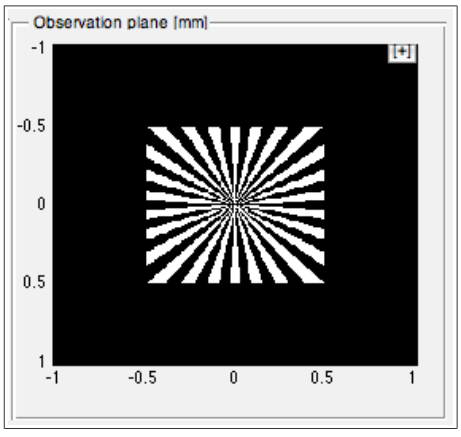

Figure 62: Observation plane

6.1.4 Observation plane (top-right)

The observation plane shows the reconstructed image after propagation, depicted as an intensity profile (amplitude-squared). The only option here is to to expand the image, using the [+] icon in the top right corner. Figure 62 depicts a reconstruction of a radial spokes test pattern in the observation plane.

6.2 FOURIOT Simulation Results and Use-Cases

There are many different combinations of parameters, options and use-cases for this tool. This section give an overview of some typical use-cases for the exploration of various scenarios that are relevant to Fourier Optics research.

Figure 63 shows a 1 mm Gaussian voxel on a 8 micron one million (1024x1024) holopixel hologram plane (4x4 mm) with almost 54k points that is propagated and reconstructed at a distance *z0*=100 mm. Fresnel angular spectrum propagation is used for hologram plane computation and observation plane reconstruction. The computation time for 1M holopixels is 0.35 seconds and the SNR is a healthy 140 dB, Figure 63. Note the source and observation images are inverted for convenient presentation on print.

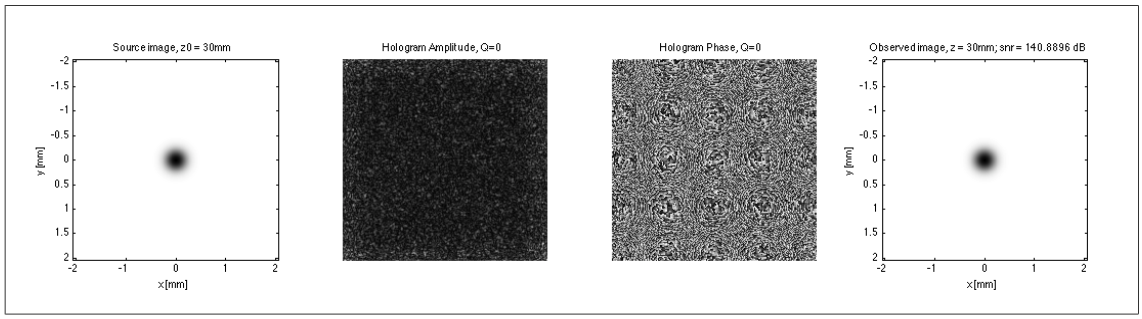

Figure 63: Gaussian, full complex implementation

The amplitude-only implementation takes real part of the complex hologram where the SNR drops dramatically to -2.2 dB, depicted in Figure 64. the reconstructed voxel is hazy but the original shape is still discernible.

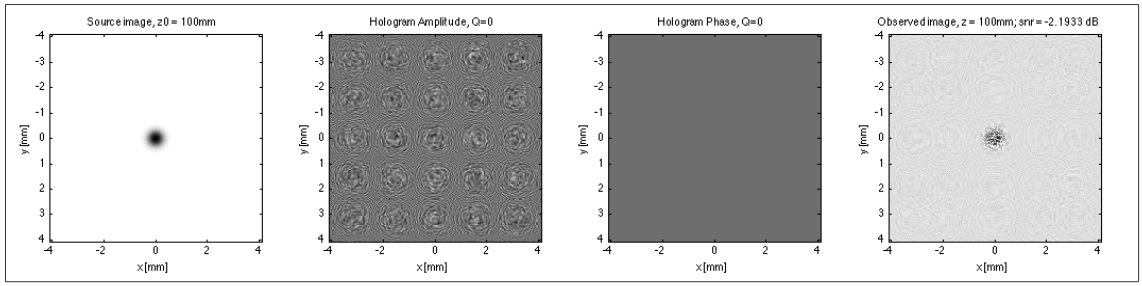

Figure 64: Real-only implementation

Next, taking the phase-only part of the hologram plane, Figure 65, the resulting SNR is 8.6 dB, which is much better than the amplitude-only version. The reconstruction is is also better defined than the amplitude-only version without the background haze.

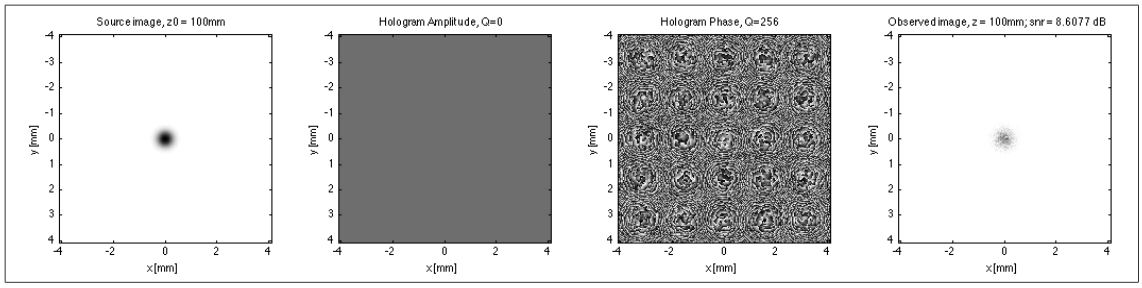

Figure 65: Phase-only implementation

Next, quantisation is applied to this phase-only implementation. Applying 256 phase levels does not have any significant impact on the SNR, which remains at 8.6 dB, and the reconstruction remains the same. This confirms the assertion that 256 phase-levels is equivalent to a continuous profile for practical purposes. Applying 16 phase levels brings the SNR down to 8.2 dB, which is still quite good and in agreement with theory where 16 phase levels have a diffraction efficiency of 98% [105, p. 72]. The impact of quantisation on the SNR for the various quantisation levels is shown in the SNR column of the table below:

Levels	SNR (dB)	Iterative SNR (dB)	Diff
None	8.6	-	-
2	1.0	1.4	0.4
4	3.4	4.4	1.0
8	7.0	10.4	3.4
16	8.2	13.9	5.7
256	8.6	15.8	7.2

Table 4: Phase only SNR with iterative optimisations

The next step is to see the impact of Iterative Fourier Transform optimisations on the SNR via the 'Iterative FT' button (section 6.1.2) for various quantisation levels, shown in the 'Iterative FT' column of Table 4. The last column of the table shows the improvement in SNR. For binary phase, there is a marginal improvement of 0.4 dB, but this increases to 7.2 dB for 256 levels. It is notable that the improvement is 3.4 dB, or over twice as good for eight levels, which is interesting for eventual DOE implementations.

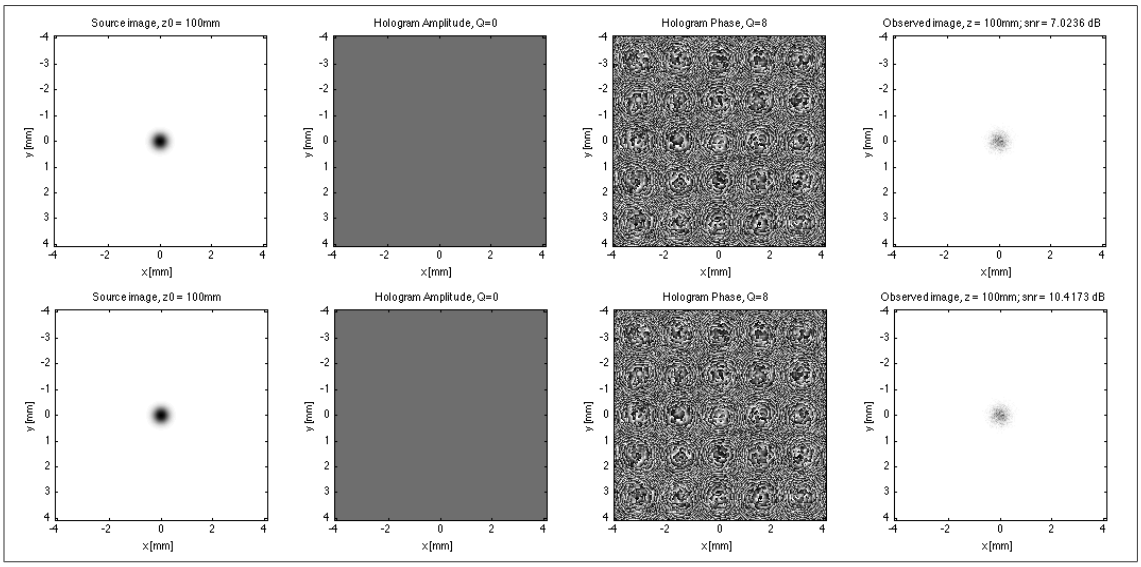

Figure 66: Phase-only with 8-levels (above) and ITFT optimisation (below)

Figure 66 illustrates the eight-level quantisation case with and without the Iterative optimisations. It is apparent that the ITFT optimisation not only improves the SNR by 3.4 dB but the resulting observation plane image also looks smoother and more 'Gaussian'-like. For this image and propagation parameters, the ITFT algorithm requires about 22 iterations and it takes over 10 seconds to perform.

6.2.1 Comparison of propagation models

The tool allows a comparison of SNRs for Kirchoff and Frensel models for both impulse response and angular spectrum propagation. This is shown in the table below:

SNR	Kirchoff (dB)	Fresnel (dB)
Impulse Response	139.8	139.7
Angular Spectrum	140.1	140.1

Table 5: Comparison of SNRs for propagation models

For the impulse response case, there is only a 0.1 dB difference between the Kirchoff and Fresnel cases. The angular spectrum propagation model is identical for both Kirchoff and Fresnel, which shows that the two models are equivalent at 100 mm. It is apparent, that, at a propagation distance of 100 mm, there is very little difference between the various models.

6.2.2 Field at different propagation distances

The *propagation* panel (section 6.1.2) allows the reconstruction distance, z, to be changed across a range of values in discrete steps. Here, we model the observed field

for a phase-only Gaussian source with z0=100 mm (using Fresnel angular spectrum model with iterative optimisations) for z at 90, 95, 100, 105, 110 and 120 mm. The behaviour of the field at these distances is depicted in the figures below.

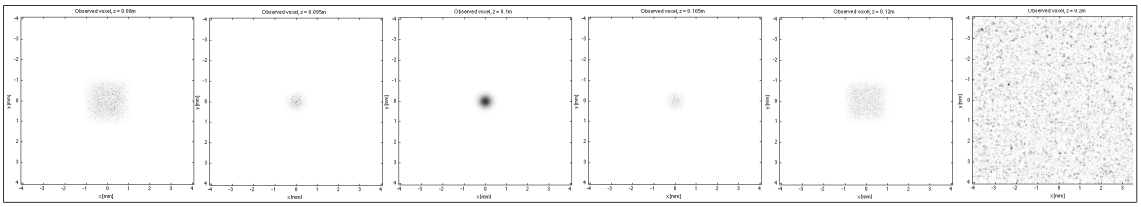

Figure 67: Field at various distances

The field expands at either side of the nominal value (z0=100 mm) and it is symmetric around this value. The field appears to retain its shape at ±5 mm (z=95, 105 mm) and tends to diminish more rapidly at ±10 mm (z=80, 110 mm) whereas it is completely indiscernible at +100 mm (z=200 mm).

6.2.3 Free space impulse response

The impulse response of free space can be seen by selecting an impulse as the source signal with the impulse response propagation model. This is depicted in Figure 68 where the phase plot resembles a classic Fresnel Zone plate; note that the impulses in the source and observation planes are not visible at this scale due to printing resolution, so this area is zoomed (1.4x) and highlighted.

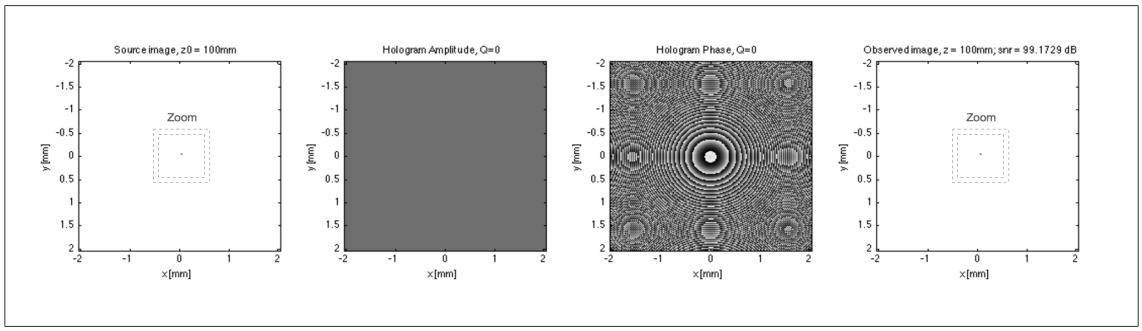

Figure 68: Impulse response

Although Angular Spectrum propagation is mathematically identical to the impulse response case, there are some practical limitations that lead to significant differences with the discrete implementation. The main issue is that the continuous-frequency chirp signal is not band-limited, so it is prone to aliasing artefacts which fold back into the band-limited spectrum imposed by the Discrete Fourier Transform. This can be seen in the amplitude and phase plots in Figure 69, which is also zoomed and highlighted. In particular, the amplitude plot is no longer uniform and the phase plot has a grainier

structure. This issue has been recognised [106,107] and tackled by band-limiting the continuous spectrum using windowing techniques commonly found in digital Fourier design [108]. These improvements may be applied in future versions of the tool.

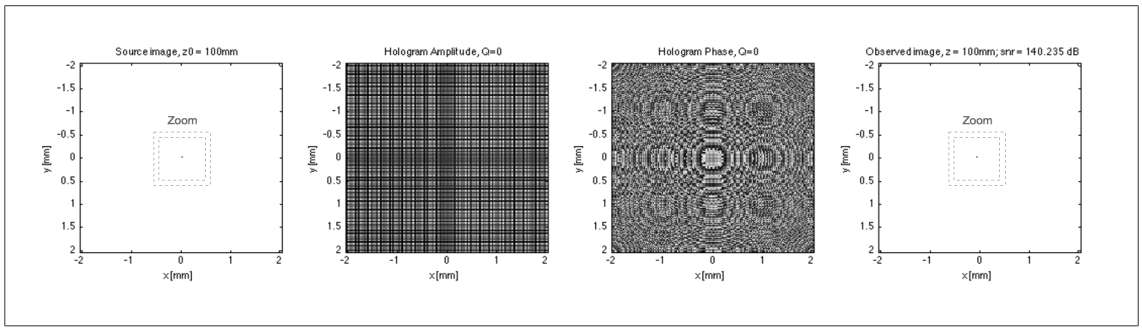

Figure 69: Angular spectrum impulse response

6.2.4 Practical SLM implementation

An SLM implementation of a seven-segment digit (source plane | shape option) is illustrated in Figure 70, using the Holoeye LCR 2500 pre-set, where the amplitude of the hologram plane is set to unity and the phase is quantised to 256 levels.

The phase pattern (second image from the right) can be exported as bitmap (section 6.1.3), which can be directly read by the Holoeye software, programmed on to the SLM, so the resulting observation plane image is viewed on a screen. This, together with a number of other source images was verified on the LCR 2500 SLM to obtain images similar to Figure 47 with a red HeNe laser (632.8 nm).

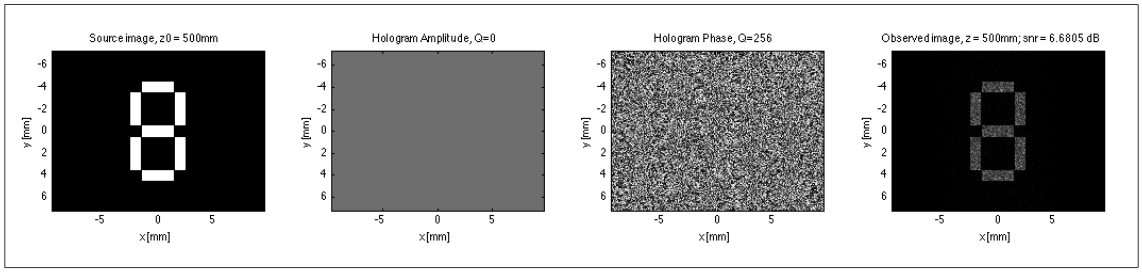

Figure 70: Seven-segment image on SLM

The seven-segment digits can be expanded (source plane → expand 1x2 option) to show two digits, Figure 71. This source image was also verified physically using a PLUTO-NIR2 SLM with a green laser (532 nm), Figure 75 in the next chapter.

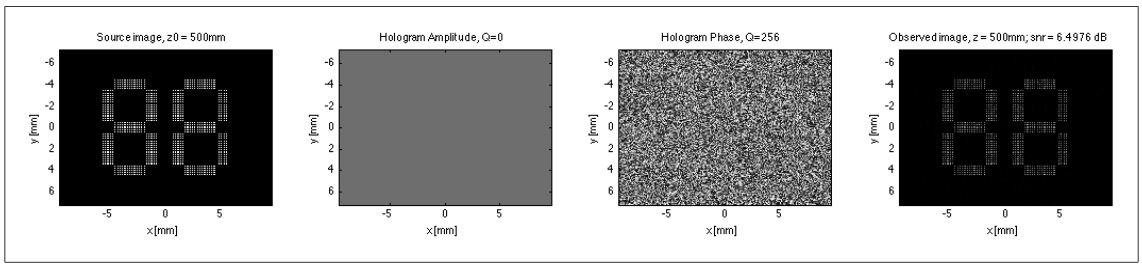

Figure 71: Expanded seven-segment digits

Note that it is unlikely that any successive runs of a design will have exactly the same set of figures shown here for SNR or other parameters, because a new set of random phase parameters is computed for each run. However, the numbers are of the same order and there are no significant variances between runs.

6.3 Implementation Notes

The FOURIOT Matlab routines, algorithms and libraries comprise over 4,000 lines of code (or over 7,000 lines including comments) with over 170 individual functions. A great deal of time was spent on optimisation of the critical algorithms and testing them in isolation before integration into the tool. Appendix D presents an overview of the main Matlab library routines used within FOURIOT and other parts of this thesis. In many cases, there is a direct correspondence between a FOURIOT action (button, slider etc) and the underlying Matlab function.

This section does not go into the details of the actual code and layout or structure of the program, rather the focus is on some of the issues faced in the implementation of the tool. Further information on the background algorithms can be found in Appendix D.

It was noticed early in the development cycle that the use of single precision arithmetic was reasonably acceptable for the impulse response and angular spectrum propagation models. This leads to a significant improvement in speed and memory size required to hold the main arrays. The FFT-based methods are generally faster and less onerous on computation and memory than convolution methods.

Although FOURIOT is not a full-blown industrial strength tool, it is stable enough and usable for research purposes. The tool was developed on a Mac computer but the Matlab interpreter can work across Linux or Windows.

6.3.1 Installation

The tool can be installed by unzipping the compressed archive into a directory. All of the required functions are located in the same directory. There are two sub-directories '*srcimages*' and '*bitmaps*' that contain some source images and output bitmaps for

loading on to SLMs.

6.4 Conclusions and further work

FOURIOT has a reasonable amount of error checking and is fairly robust, however, it can occasionally get thrown off-track and hang or crash. This usually happens when it is in the middle of a large calculation and it is suddenly asked to do something else. The tool does nothing by way of checking or constraining the parameters of the source images. Hence it is possible for the user to construct all sorts of strange shapes which may even exceed the limits of the source image resolution. This occasionally lead to unstable behaviour such as hanging or crashing the program. Future versions could include constraints to control this.

The use-cases enable exploration of the design space for both DOE and SLM implementations (phase and amplitude) across a multitude of parameters (pixel pitch, hologram size, distance, quantisation, propagation type, optimisation). Apart from a full set of default images, any arbitrary source image can be uploaded. The benefit of the tool is almost immediate visualisation for trying out different scenarios given the wide range of possibilities within the design space. It can also produce outputs that can subsequently be shown on an SLM or eventually fabricated as a DOE to verify designs, explore algorithm implementations and fine-tune parameters. As far as I am aware, this is a comprehensive tool for Fourier-Optical propagation and analysis.

The tool is designed to be highly modular with expansion in mind. Hence, it is relatively easy to extend current options or introduce new features. For example, the addition of a profile for a new SLM was done in less than an hour.

The work presented here is a starting point and there is plenty of scope for further enhancements or optimisations in terms of functionality and usability. Optimisations could include the following:

- Faster computation: use $R^2=sqrt(X^2+Y^2)$ instead of meshgrid X,Y co-ordinates
- Faster computation & larger arrays: use parallel computing/GPUs
- Make a binary executable for speed-up and distribution

The following features could be added to improve functionality:

- Implement band-limited angular spectrum propagation to prevent aliasing artefacts [106,107]
- Quasi-coherent and incoherent illumination models

- Fourier Optical propagation models

The following would enhance usability of the tool:

- Add comprehensive help, including a diagram of the propagation model
- Save a 'source object' option
- Make the tool available to the research community for feedback/suggestions
- More extensive testing, test suite/sequences for quality improvements

Although inspired by a number of recent books on the topic [96,109,110], this work is entirely my own.

Chapter 7 - Spatial-Multiplexing Computer Generated Holograms

Given the challenges of DOE fabrication and the disappointing results, it was decided to proceed with the CGH work in the simulation domain using FOURIOT and other code derived from it. The work here looks more closely at the principles behind the spatial-multiplexing and sampling techniques demonstrated in Chapter 2, which were based on empirical approaches. The spatial sampling and interleaving techniques are now underpinned by Fourier Optical models from Chapters 4 and 5. The models are first implemented in a simulation environment that can produce images for presentation on an SLM for physical verification and measurements. Although this is not quite the same as DOE fabrication, the SLM approach has considerably lower cost and risk as well as a faster turnaround. The design principles and implementation of SLMs and DOEs are very similar. The idea is to recreate imagery made on the initial display based on classic analogue holography from Chapter 2 for a set of segmented digits.

The initial investigations for this work were performed within the FOURIOT simulator from the previous chapter. One of the default source images includes segmented digits which were studied with respect to quantisation, resolution, bandwidth, SNR, sampling period etc. The code for this work is implemented as a series of separate Matlab programs, which use the FOURIOT library (Appendix D) and extend it in some cases. A holographic screen containing a set of multiplexed spatially-interleaved interference patterns is computed, simulated and programmed on a Spatial Light Modulator (SLM) for verification. A secondary level of sampling is performed over and above the baseline resolution of the DOE or SLM.

7.1 Sixteen segment image design

The holographic screen contains up to sixteen spatially-multiplexed interference patterns. Each pattern is capable of reconstructing a single segment of a double-digit seven segment display together with ± symbols, Figure 72. Each of the sixteen segments is denoted by a letter: from 'a' to 'p'. The least significant digit is represented by the segments 'a' to 'g', the most significant digit by 'h' to 'n' and the ± symbols by 'o' and 'p'.

The specifications of the Holoeye NIR-2 SLM device are:

- Full HD resolution 1080x1920 pixels (8.6x15.4mm)
- Pixel pitch of 8 microns
- Reflection mode phase modulation
- 256 linear phase levels (8-bit modulation)

The segments are designed to fit within the 8.6x15.4 mm geometry of the SLM, so the size of a single segment is 2.5x0.83 mm (Figure 72). For this design, the propagation distance was selected to be 0.5 m.

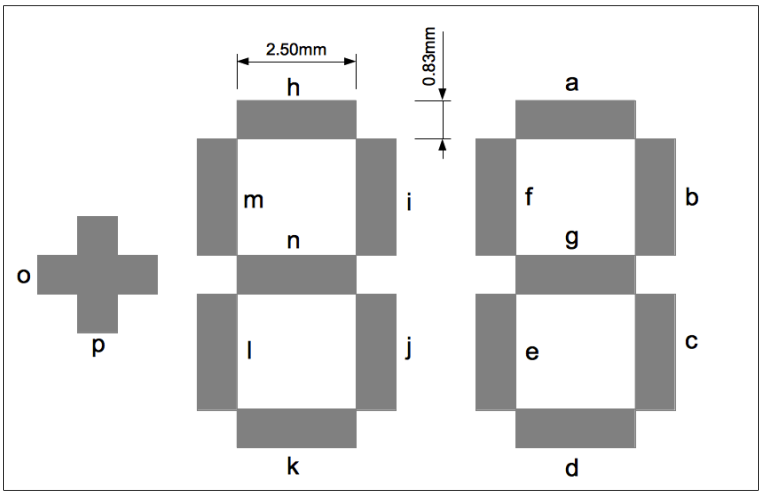

Figure 72: Sixteen segment design

7.2 Holographic screen composition

A methodology for determining the bandwidth and sampling requirements of a finite signal is presented in section 5.2. The Fourier Transform of the signal is constrained to encapsulate 98% of the total signal energy, thereby yielding an upper bound on the sampling interval. Considering a single 2.5x0.83 mm segment in one dimension, then the portion of this segment with the highest bandwidth is 0.83 mm. It is shown in section 5.2.2 that a 1 mm segment has a bandwidth of 5220 m^{-1}. Accordingly, the single-sided bandwidth B of a 0.83 mm segment is given by equation (63):

$$B = \frac{5.22}{a} = \frac{5220}{0.83} = 6274\, m^{-1} \tag{72}$$

The minimum sampling interval is given by taking the reciprocal of the full double-sided bandwidth, in conformance with the sampling theorem, from equation (73) below. Of course, this is now represented in the digital domain via the DFT, where the spectrum is no longer finite but repeats according to the sampling interval.

$$\Delta x = \frac{1}{2B} = 79.6\, \mu m \tag{73}$$

Hence the minimum sampling requirement for this signal is almost 80 μm, so at least eleven samples are required to represent the 0.83 mm segment. Given that the SLM

pixel pitch is 8μm, we have an oversampling factor of about nine times. Hence, at 8 μm, we have anything from 11 to 100 samples with which to reconstruct the segment. The lower limit of eleven samples is insufficient because it represents critical sampling that would require an ideal bandpass filter in the frequency domain with a bandwidth of $\pm B$. A non-ideal reconstruction filter would lead to a high degree of aliasing. The upper limit of a hundred samples provides headroom to perform spatial multiplexing. A reasonable starting point for the number of samples is midway between these values e.g. 60 samples. The spatial-multiplexing or amplitude masking enables particular sets of samples to be selected for reconstruction. This scheme can be viewed as a form of spatially-selective filtering.

The holographic screen comprises a macro-pixel structure across the SLM, arranged as 4x4 sub-pixels or sixteen sub-pixels per macro-pixel. Each sub-pixel in the hologram

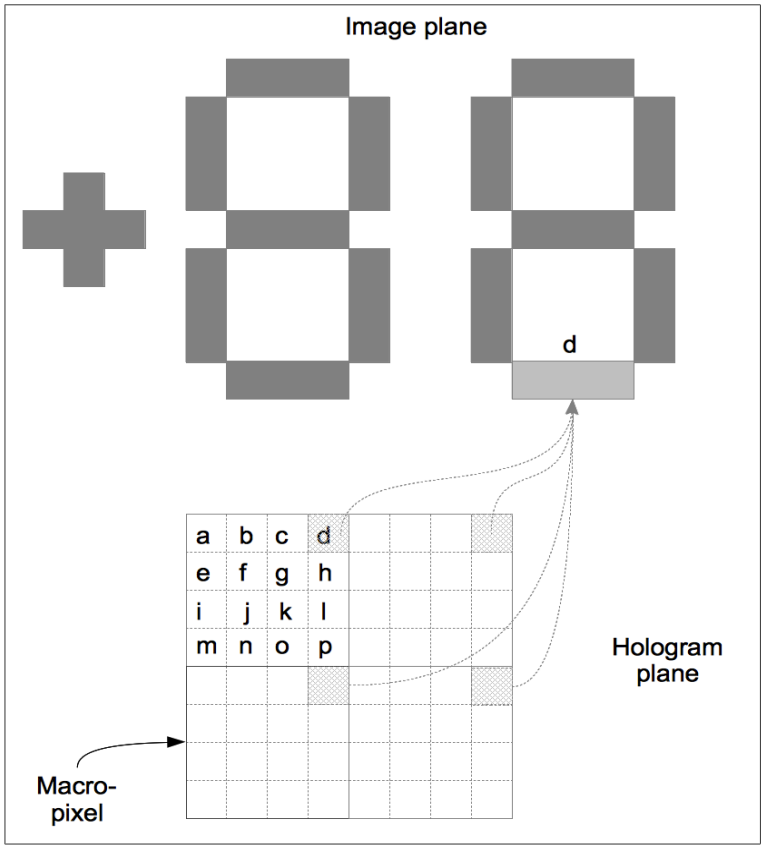

Figure 73: Mapping segments to macro and sub-pixels

plane maps on to a single segment in the image plane. Figure 73 illustrates the segmented digits in the image plane and a subset of the macro-pixels in the hologram plane. The sub-pixels are labelled 'a' to 'p' and each of these makes a contribution to its corresponding segment in Figure 72. Figure 73 highlights the contributions of the sub-pixels 'd' across several macro-pixels to the corresponding image segment 'd'.

The segments are generated using the sevenSeg and nineSeg Matlab functions from the FOURIOT library (Appendix D). It is possible to use any number of propagation models to compute the hologram plane, from direct convolution to the impulse response or angular spectrum methods discussed in Chapters 4 and 5 using the FOURIOT simulator from Chapter 6. For this design, at 0.5m, Fresnel angular spectrum is appropriate, which was verified with FOURIOT.

The Matlab code for the (simulated) holographic display is in two parts. The first program *sixteenSegFF.m* (sixteen segments fixed-frame) computes the hologram plane patterns for all sixteen segments via Fresnel angular spectrum propagation with iterative Fourier Transform optimisation. These patterns are masked and interleaved to make the composite holographic screen comprising the macro and sub-pixel structures. The mask patterns are generated by a mask generation support function: generateMasks.m. Design optimisation is performed via an enhanced Iterative Fourier Transform algorithm which takes amplitude masking into account. This is implemented as a Matlab function ITFT3, building upon the earlier ITFT2 algorithm (Section 5.3.7 and Appendix D). The optimisation process harmonises the phase distribution across the segments and improves the SNR. The ITFT3 algorithm improves the SNR for the individual segments by around 0.5dB on average. Computation of the composite hologram with these optimisations takes about five minutes.

The second part of the simulator *replay16seg.m*, performs the reconstruction (reverse propagation). The program takes the holographic screen generated by *sixteenSegFF.m* and applies selective masking to reproduce sequences of patterns corresponding to recognisable seven-segment numerals. For example, the numeral '2' can be reproduced with the least significant segments 'a', 'b', 'd', 'e' and 'g' (Figure 72). These segments are selected in the hologram plane by masking off the sub-pixel patterns for all other segments i.e. 'c', 'f' and 'h' to 'o'. In this way, the program maps the seven-segment representations of all ten numerals (i.e. zero to nine) in both positions (most and least significant positions) as well as the plus and minus symbols. The program can therefore sequence or count through any combination of numbers from -99 to +99.

The Matlab simulator has two options to perform the masking: the simplest and most obvious is amplitude masking which applies zero-values to those sub-pixels that are not required. The second option puts random phase values into these sub-pixels instead; the reason for this is that the SLM is configured as a phase-only device so it is not capable of performing the amplitude modulation required for masking. The impact of this

alternative random-phase masking approach is on the quality of the reconstructed image. It is expected that the image would be less uniform and fuzzy due to additional noise contributed by scattering caused by the random-phase mask.

7.3 Results and observations

This section reports both simulated and experimental results. The phase pattern for the composite interleaved holographic screen computed for the SLM geometry (section 7.2) is shown in Figure 74. The pixellated structure of the sub-pixels can be seen in the figure. The sub-pixel size is 60x60 samples, which makes the macro-pixel 240x240 samples; at 8μm i.e. 1.92x1.92 mm. A number of simulations were performed with sub-pixel values varying between 20 and 70 samples. Aliasing was observed around the lower end of the scale, around 20 to 40 samples. The simulations yielded visually good results without aliasing or other artefacts with 50 samples or above, hence the choice of 60 samples is validated.

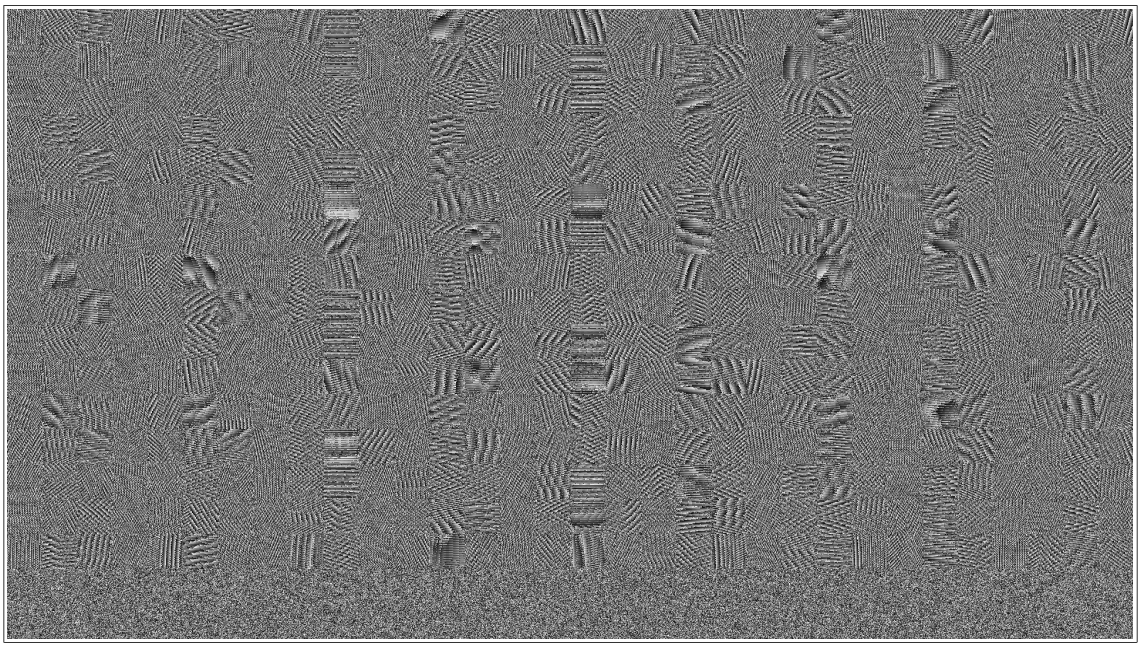

Figure 74: Composite interleaved hologram pattern

The experimental setup is shown schematically in Figure 75, where a collimated green laser source illuminates the reflective SLM, or hologram plane, that is imaged on the diffusion screen for observation. The SLM behaves as a secondary screen for a computer that displays the phase pattern as a bitmap. The SLM is calibrated to represent a zero to 2π phase shift by a greyscale level between black (0) and white (255).

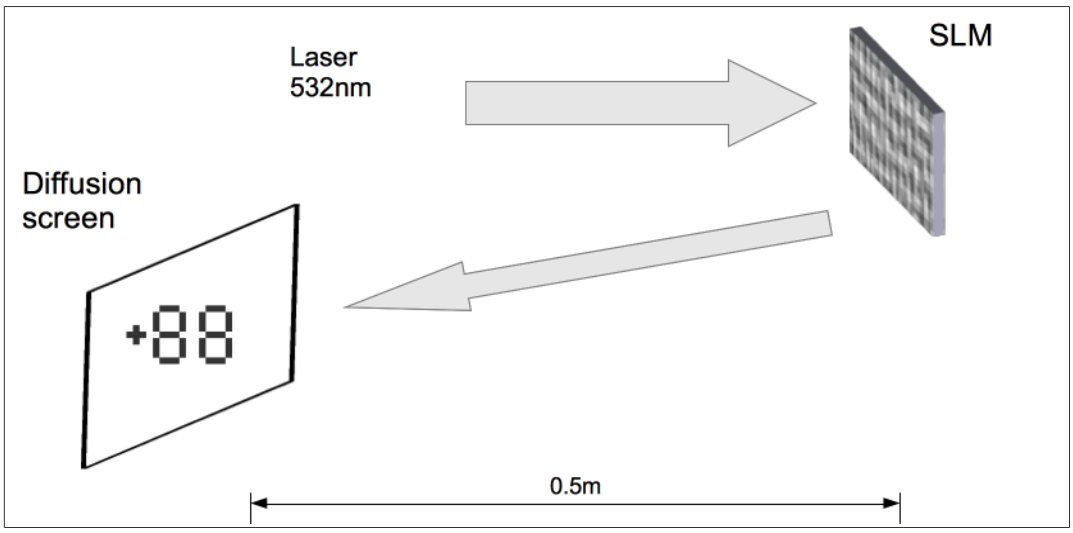

Figure 75: Experimental setup with Holoeye SLM

The simulator program *replay16seg.m* was used to generate the masked holographic screen patterns corresponding to the numerals +88 (all segments illuminated), +53 and -12. These are depicted in Figure 76, which shows the masked holographic screen on the left, simulated reconstruction in the middle and a photo of the reconstructed image on the diffusion screen on the right.

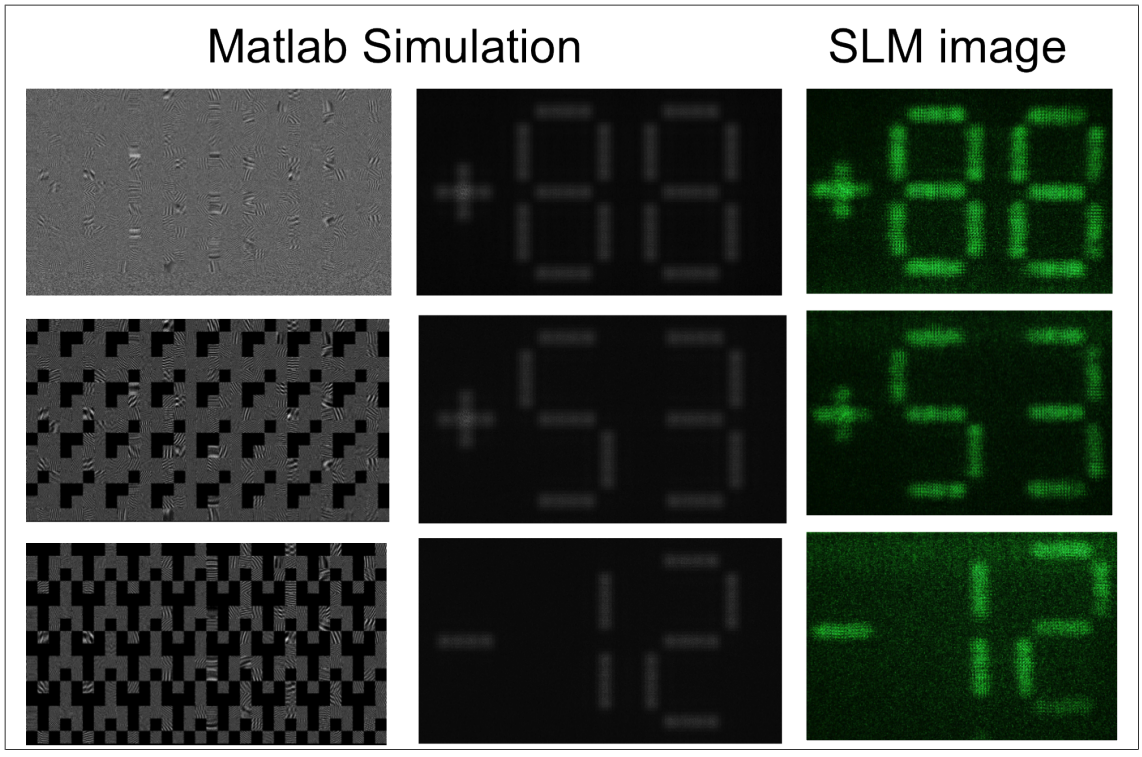

Figure 76: Simulated and actual results

The topmost row of images corresponding to the digit +88 does not have any masking as it exposes all image segments. Visually, the uniformity of the segments is good and the central part of the 'plus' symbol is brighter than the edges, in both simulated and

actual results. This is because the cross is composed of two vertical lines and they interfere constructively in the central overlapping area. The middle row shows four segments masked off to produce +53. Again, the uniformity is good and dark areas of the image even appear to have less noise than the the previous digits. This may be due to the way the camera sensor has integrated the pixels. The last row in Figure 76 has eight segments masked off to depict '-12'. Although the simulated result is consistent with the other simulated images, the SLM image appears to be noisier than the previous images whereas the lowest segment (segment d in Figure 73) of the least-significant digit appears to be less uniform. This is most likely due to noise from increased scattering by the large number of phase-masked segments.

Although Figure 76 shows amplitude-masking, in fact, the Matlab simulator was configured in random-phase mode and the simulated results are shown for this masking mode. The left-hand plots show amplitude masking simply for clarity as the mask is clearly discernible in black, whereas a random-phase mask would not be visibly distinguishable. Note that, apart from some cropping, there was no other signal processing applied to the photographs.

7.4 Conclusions and further work

The basic principles behind the sampling and interleaving approach were demonstrated in Chapter 2, based on classical interference holography. The work presented here validates the spatial multiplexing and interleaving approach and provides some hints on the properties and limitations of this approach, depending on the spectrum of the segments. CGH enables full end-to-end control of the holoscreen and is amenable to diffractive optical element (DOE) fabrication via photolithography (sections 1.4.2 and 4.2).

The display system is first simulated in Matlab, including computation and optimisation of the interference patterns for the holoscreen and reconstruction. The algorithms produce diffraction patterns that can be shown directly on the SLM for measurements and validation. There is good qualitative agreement between the simulated and experimental results. This work was presented at the Liquid Crystal Photonics conference, Hong Kong in December 2012.

It would be useful to investigate the reconstructed images more closely using Matlab image processing algorithms to isolate segments and assess them according to some metrics to extract parameters such as uniformity, SNR, contrast ratio etc.

Holograms can hold a great deal of information by virtue of a very high space-

bandwidth product. For the simple image segments presented here, this space-bandwidth product is too high, leading to a great deal of information redundancy, a feature that can be exploited to realise practical holographic displays. Fully dynamic holographic displays have been difficult to realise in practice (section 1.2.5) but this approach shows that, by applying some simplifications, it is possible to achieve real-time dynamic holographic displays. Such displays would be limited to symbology or segmented numerals that can be found in many consumer and professional applications. For example, a realtime holographic clock could be realised by extending this work from sixteen to twenty four segments and implementing the hologram on a DOE instead of an SLM, where the DOE can be mass-produced in plastic via embossing. Masking can be provided by a simple passive-matrix LCD, as shown in Chapter 3.

The main advantage of this approach is simplicity where the holographic screen is pre-computed. Switching between the segments is performed by mask-programmable structured illumination of the holographic screen without the need to recompute the diffraction patterns. The work has also extended the total number of addressed segments from nine in Chapter 2 to sixteen. As previously shown (Chapters 2 and 3), any segment or combination of segments can be selected in any combination to make the numerals.

Classical holographic recording introduces unwanted artefacts, in the form of pseudoscopic twin-images (section 1.3). The undesirable conjugate image can be removed with CGH. Also, classical holography requires off-axis recording and replay geometries that can sometimes lead to awkward optical architectures. CGH is capable of making on-axis holograms and reconstructions, thus easing optical design.

The work shows that phase masking is an alternative means to perform selective reconstruction. This could have practical significance by, for example, using a phase-scattering mechanism such as polymer-dispersed liquid crystal (PDLC) devices instead of amplitude-modulated LCDs. This could lead to low-cost and readily implementable displays.

Although this work used regular sub-pixel arrangements, this is done for simplicity and convenience but it is not an absolute requirement. It is possible to arrange the sub-pixels in any order, including random arrangements, although this makes the masking process somewhat more complex for both recording and reconstruction. The segments presented in this work had point spacings the same as the SLM holopixel width of 8 microns. Again, this was done for convenience. The human eye is not capable of discerning such closely-packed pixels and it is possible to have the image points spaced

wider apart. Although the image size was constrained to fit within the area of the SLM, this is not a strict requirement as it is possible to make the segments larger than the area of the SLM. Possible approaches to achieve this include direct convolution methods where the point cloud representing the segments cover a larger area; or through the use of magnification in Fourier Optics by employing an intermediate plane [96, p. 199].

Chapter 8 - Holographic Optical Elements and Interactive Display

Holographic displays are usually portrayed in the popular media by special effects generated by computer graphics. More commonly, holographic displays are mis-represented by optical illusions such as Pepper's Ghost effects [15, p. 249,111]. Such illusions are often virtual images behind a screen or semi-reflective surface, which inhibits the interactivity of any display system based on this approach. A key goal for holographic display systems is the creation of real-space images that can be updated in real-time that enables direct human interaction with the holographic images [112, p. 34]. The aim of this thesis work is to realise such a display with techniques that are viable and compatible with current technology.

The first three generations of displays are based on a holographic screen containing a series of pre-encoded interference patterns. The encoding is done with spatial sampling and multiplexing techniques that enabled several distinct interference patterns to be stored in the same holographic screen. The images can be reconstructed in any order or sequence in real-time simply by rear-illuminating the holographic screen with different patterns of light generated from a projector or using a mask. The main problem with the approach taken so far is scalability which limits the resolution of the display to some tens or maybe up to a hundred voxels. As the amount of sampling or voxels increases, the amount of light passing through the hologram decreases due to the increased number of mask patterns. This implies a reduction in the brightness of the reconstructed voxels, so it is necessary to increase the illumination to compensate. This could lead to safety issues for laser-illuminated systems. These limitations may be acceptable for some simple applications but it remains a severe constraint for widespread adoption. This chapter presents work directed to overcoming such issues on scalability.

The research across various generations of display has led to ever-smaller holographic elements or hogels with classical holography and holoprinters (millimetre dimensions) on the one hand and fine holopixels on DOEs (tens/hundreds micron dimension) on the other. In the work presented thusfar, the focus has been on encoding and reconstructing arbitrary single voxels in (x,y,z) space. The attention of this work turns to holographic optical elements that can perform entire optical functions (next section) and attempts to use some of the features to further reduce complexity and increase voxel resolution. The motivation behind the work presented here is to overcome limitations of the previous displays, enabling higher resolution and arbitrary images to be created as well as introducing new interactive functionality.

8.1 Overview of Holographic Optical Elements

DOEs have been discussed extensively in earlier chapters. These are part of a wider category of generalised Holographic Optical Elements (HOEs) that perform specific optical functions instead of forming near or far-field images. Such functions include but are not limited to:

- Lenses (converging or diverging)
- Mirrors (planar, spherical, elliptic)
- Spectral and spatial filtering
- Prisms
- Beamsplitters (fan-outs)
- Gratings
- Interconnects

HOEs may include complex combinations of these functions with the potential to simplify large optical setups. HOEs can be made using all of the techniques described in this thesis, including classic interference holography, holoprinting, embossing, photolithography (DOEs) etc. Accordingly, HOEs can be made in transmission or reflection mode, depending on the functionality required [68, p. 217,113]. There are a number of advantages to using diffractive optics compared to conventional refractive optics including: thinner optical elements with associated reduction in size and weight as well as potentially lower cost (for larger volumes). The disadvantages of diffractive vs refractive elements are: constrained operating parameters i.e. the system is usually designed for specific wavelengths or angles of incidence, which make it sensitive to manufacturing tolerances. Transmissive diffractive elements have an inherent chromatic dispersion characteristic that can negatively impact system performance. Apart from certain applications such as spectrometers, this chromatic dispersion property is often regarded as an aberration by optical designers, who usually attempt to design it out of an optical system.

8.2 Converging HOE lens theory

For this work, the most attractive feature of the HOE is that it can be fabricated with a large aperture (>10 cm) and a short focal length, which is a similar dimension to the aperture, corresponding to an f/# of around unity. The display is designed for an aperture size of 20x25 cm. The HOE is a converging transmission diffractive lens. Larger apertures could be realised using conventional refractive optics but the lens

would be very bulky and costly. The situation could be improved significantly by employing Fresnel structures, however, Fresnel components are generally used in bulk optics for light collimation type applications and tend to have poor image quality when used for imaging.

The HOE is made using classical off-axis interference holography (section 1.3.1) where a plane reference beam inclined at an angle θ to the hologram plane interferes with a field due to a point source at a distance f_0 from the hologram plane. The recording setup is depicted in Figure 77.

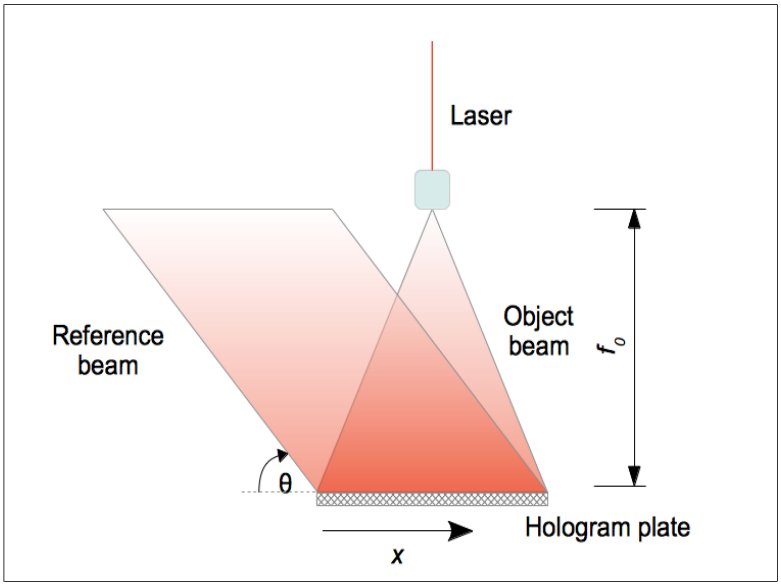

Figure 77: HOE recording setup

The electric field component of the reference beam Er is given by equation (74) along the horizontal *x*-axis (see Figure 77), where *k* is the modulus of the wave-vector:

$$E_r = \exp(jkx\sin\theta) \tag{74}$$

The electric field due to the point source E_p at a distance of f_0 from the hologram plane (Figure 77) is given by the Fresnel approximation (Appendix C) of a spherical diverging beam by a quadratic field via equation (75) representing the object beam:

$$E_p = \exp\left(\frac{jk}{2f_0}x^2\right) \tag{75}$$

The wave amplitudes are set to unity in equations (74) and (75) for simplicity, whereas practical beams have Gaussian-like profiles. The holographic plate records the intensity of the interfering waves, which must be mutually coherent and have the same

polarisation. Recall from equation (4) that the total intensity is given by the complex sum of the plane wave and point source object:

$$I = |E_r + E_p|^2 = |E_r|^2 + |E_p|^2 + E_r E_p^* + E_r^* E_p \tag{76}$$

The first two terms in (76) are intensities of the reference and object waves, known as constant bias terms where $|E_r|=|E_p|=1$ that only affect the hologram recording process (section 1.5). However, these bias terms do not contain object information in terms of amplitude or phase, so they may be ignored. Substituting (74) and (75) in (76) and expanding:

$$I = \exp(jkx\sin\theta)\exp\left(-\frac{jk}{2f_0}x^2\right) + \exp(-jkx\sin\theta)\exp\left(\frac{jk}{2f_0}x^2\right) \tag{77}$$

Accordingly, equation (77) represents a pair of diverging and converging optical elements. The first term is responsible for the converging lens behaviour. The second diverging term is undesirable but it is inherent in classical holography and is conveniently spatially separated due to the off-axis recording configuration (section 1.3). Equation (77) can be simplified further:

$$I = \cos\left(2\pi\Lambda x - \frac{k}{2f_0}x^2\right) \tag{78}$$

Where Λ is the average fringe frequency, from equation (1). Equation (78) is a real function of x, showing how the object beam modulates the phase of the fringes relative to the "carrier" frequency Λ. This real intensity function exposes the holographic material (section 1.5) to produce an amplitude hologram, after it is developed chemically. It can be shown that the maximum diffraction efficiency of a transmission amplitude hologram is <33% [53, pp. 59–60,114, p. 61], because such a hologram both absorbs and transmits light. If the hologram is encoded as a pure phase hologram, then the diffraction efficiency can be improved considerably, up to 100%, transmitting all of the light. The amplitude hologram is converted into a phase hologram through bleaching, a chemical process that converts intensity variations into refractive index variations within the holographic medium (section 1.5). The refractive index modulation depth depends on the properties of the holographic material. The resulting refractive index profile, N, is given by the argument of equation (78):

$$N \propto 2\pi\Lambda x - \frac{k}{2f_0}x^2 \tag{79}$$

Hence, the optical phase transmission profile, T, as seen by a reconstruction beam is given by equation (80), which is also extended to two dimensions by applying separation of variables expressed as a function of x and y:

$$T(x, y) = \exp j\left(2\pi\Lambda x - \frac{k}{2f_0}(x^2 + y^2)\right) \tag{80}$$

The second term in equation (80) shows that the phase transmission profile of the resulting hologram is that of a converging lens. In practice, the developed and bleached hologram is a mixture somewhere between the amplitude hologram (78) and phase hologram (80). Consequently, the diffraction efficiency is also expected to lie somewhere between 33% to 90%.

8.3 Holographic display system model and design

The optical architecture of the display is based on a projection system design. The source image produced by a laser projector is imaged on a diffusion screen that acts as an object for the converging HOE, depicted in Figure 78.

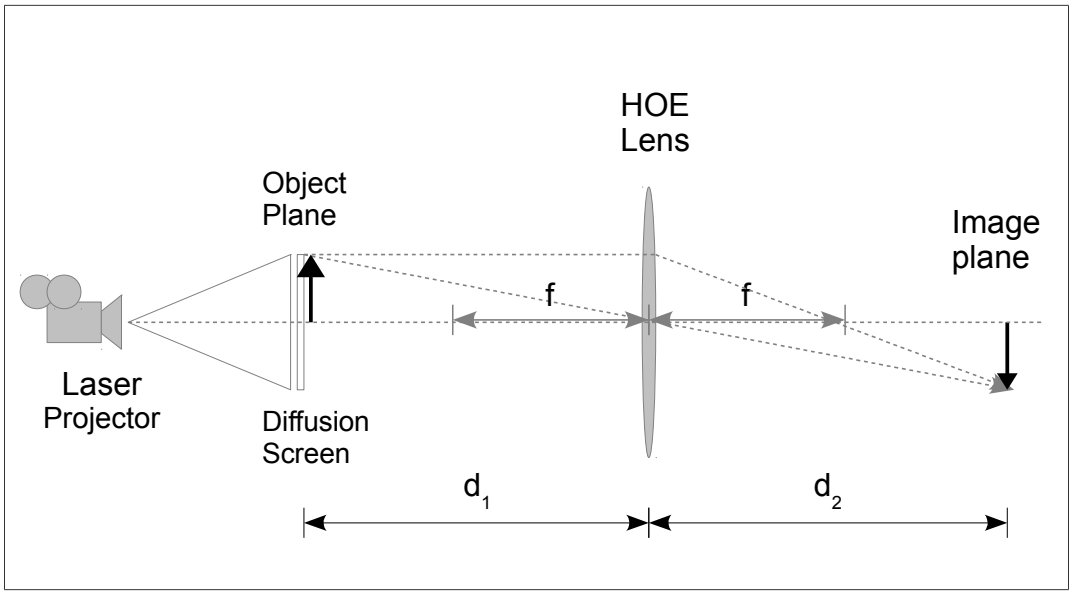

Figure 78: HOE lens with laser projection

The diffusion screen acts as the object placed at a distance d_1 before the lens, which is imaged at a distance d_2 after the lens. The optical performance of the imaging system can be described through Fourier Imaging Optics [115, p. 53]. The imaging behaviour

of the HOE is similar to that of a classic thin lens from geometric optics according to (81):

$$\frac{1}{d_1}+\frac{1}{d_2}=\frac{1}{f(\lambda)} \tag{81}$$

Where $f(\lambda)$ is the focal length of the HOE lens. However, expression (81) is only valid at a single wavelength, λ. The chromatic dispersion or shift of the focal point, f_0, with a different replay wavelength λ_r with respect to the recording wavelength, λ_0, is given by expression (82):

$$f(\lambda)=f_0\frac{\lambda_0}{\lambda_r}=\frac{f_0}{\mu} \tag{82}$$

The recorded fringe frequency, Λ, may be subject to further shifting during development and bleaching of the holographic material due to shrinkage or swelling. This leads to additional phase changes across the entire hologram, assuming the shrinkage is uniform, that results in an apparent shift in the recorded wavelength, λ_0. Equation (82) show that the focal point shifts inversely with the replay wavelength. The ratio of replay to recording wavelength is encapsulated in the μ term. It is evident that when $f=f_0$ i.e. $\mu=1$ only when both recording and replay wavelengths match. The magnification of the image is given by:

$$M=-\frac{d_2}{d_1} \tag{83}$$

The image is thus inverted, the same as geometric optics. It follows that an object placed at f is imaged at infinity whereas an object at $2f$ is imaged at unity magnification. Anything placed between f and $2f$ is magnified and will be presented in real space; objects $>2f$ are diminished (reduced magnification) and real; whereas anything placed $<f$ is virtual and on the same side of the lens. The preferred working distance for the display system is $d_1=d_2\approx 2f_0$ to obtain unity magnification at the recording wavelength. In practice, d_1 is fixed, substituting $d_1=2f_0$ into (81) and (82) yields:

$$d_2=\frac{2f_0}{2\mu-1} \tag{84}$$

Substituting this in equation (83) gives:

$$M = \frac{1}{2\mu - 1} \tag{85}$$

The distances and magnification for red, green and blue wavelengths are illustrated in Figure 79.

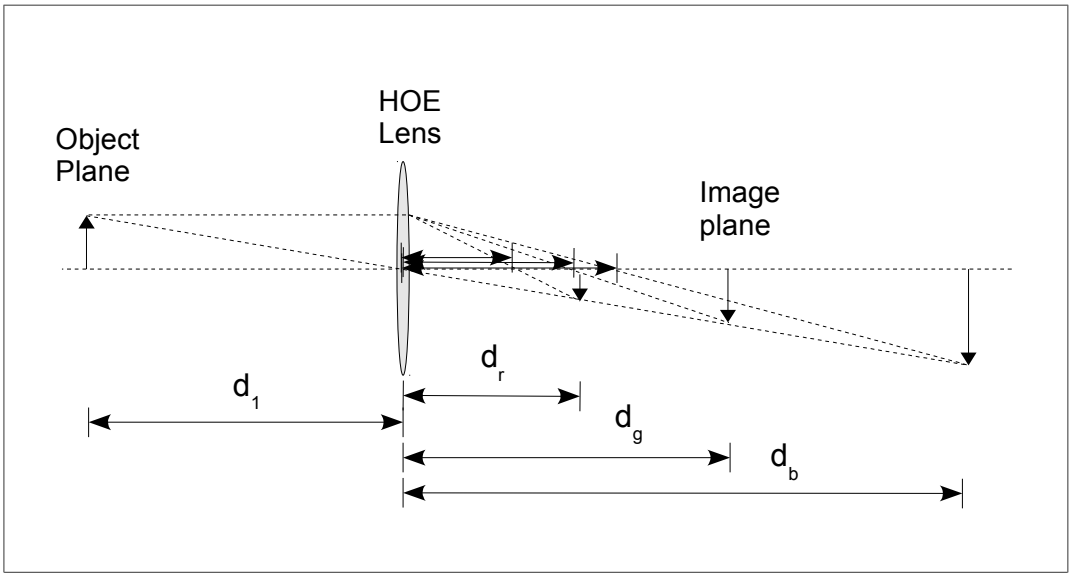

Figure 79: HOE behaviour at various (RGB) wavelengths

Figure 79 shows the difference in magnification and imaging distances due to the chromatic dispersion of the HOE lens traced via classical ray diagrams. Each of the red, green and blue wavelengths have different focii, resulting in three different image formation distances d_2 labelled as d_r, d_g and d_b. These parameters are enumerated in Table 6, for f_0=20 cm, where the shift between the red and blue planes is around 32 cm. The table also shows the values μ and magnification, M at wavelengths of 442, 532 and 642 nm (chosen for a laser projector in section 8.6).

The angle of reconstruction also shifts according to the replay wavelength, similar to the other parameters (f_0, magnification etc). This is due to the off-axis recording mechanism whereby the grating equation (1) applies to the recording wavelength and angle.

$$\Lambda = \frac{\sin\theta_0}{\lambda_0} = \frac{\sin\theta}{\lambda_r}$$
$$\theta = \arcsin(\mu \sin\theta_0) \tag{86}$$

The last row of the Table 6 enumerates the variation of reconstruction angle θ for an

initial recording angle of θ_0=20 degrees. It is apparent that there is an angular shift of 7.8 degrees between the red and blue planes.

λ (nm)	442	532	642
μ	0.8308	1.000	1.2068
M	-1.5114	-1.000	-0.7074
d_2 (d_1 @ $2f_0$) cm	60.45	40.00	28.30
θ (degrees)	24.3	20.0	16.5

Table 6: HOE design parameters

The viewing angle of the image at d_2 is given by:

$$\alpha \approx 2\arctan\left(\frac{H}{2\,d_2}\right) = 2\arctan\left(\frac{H}{4\,f_0}\right) \tag{87}$$

Where H is the size of the HOE aperture, setting H=20 cm and f_0=20 cm (i.e. f/# of unity), the expected viewing angle is 28.1 degrees.

8.4 HOE Fabrication

The holographic recording configuration is a point source generated by a microscope objective expanding a laser beam, located 22 cm from the hologram plane. This object beam is interfered with a collimated reference beam at an angle of 20 degrees to the plane of the hologram. The reference beam was generated with a beam expander placed at the focal point of a 32 cm spherical mirror in an off-axis configuration. Both beams cover the entire surface of the hologram plane. The recording configuration is shown in Figure 77.

The first set of HOEs were made using Dichromated Gelatin (DCG), a standard green-sensitive holographic material coated on a 20x25 cm glass plate. The recording setup uses a 50 mW green DPSS laser at λ_0=532 nm. DCG requires subsequent chemical processing to develop the material before being sealed by a second glass plate against the ingress of moisture and humidity to prevent degradation. A series of HOEs were fabricated in this manner with DCG.

A second HOE was fabricated with the latest Bayer Bayfol® HX green-sensitive photopolymer (PP) [116]. This material has the advantage that it is developed directly using the incident laser energy, so no further wet processing is required. This material has a sensitivity of around 30 mJ/cm^2, which is three times more sensitive than DCG (~100 mJ/cm^2), thus requiring shorter exposure time and less incident laser power. The

refractive index modulation is 0.03 with low shrinkage (<2%). The photopolymer is in the form of a thin adhesive plastic film that is easily mounted on a clear glass substrate.

The DCG and Bayer PP are phase HOEs. Although DCG is relatively well-known, it is less stable and more difficult to produce than the Bayer PP, which does not require any chemical processing. The HOEs were produced with both technologies for comparison purposes, with a view towards using PP in the longer term due to its ease of fabrication and other desirable properties. Since the Bayer PP is so new and experimental, as far as we are aware, it is the first time that such a large HOE has been fabricated with this material. At the time of writing, the availability and price of the PP is unknown.

8.5 Digital subsystems and interactivity

The remaining electronic components of the holographic display system include an embedded controller that produces the images for the laser projector and processes motion sensor data for interactivity. Apart from the optical and electronics hardware, the display system comprises extensive software elements to facilitate the coding of interactive applications via a library of functions within an Application Programming Interface (API).

The laser projector is the consumer Show-WX pico-projector from Microvision with a WVGA resolution (848x480) in a compact form factor. The picoprojector is based on a single MEMs micro-mirror scanning a triplet of RGB lasers (642 nm, 532 nm, 442 nm) with a brightness of 10 lumens. Interactivity is provided by the Microsoft Kinect motion sensor, which is based on illumination and detection of a structured IR beam which maps distances from the sensor to the detected scene as greyscale levels at VGA resolution (640x480). Although initially designed for the Xbox gaming console, the Kinect sensor is widely used in many fields of interactive research. The motion depth varies between ~0.5 to 2.0 m. The digital controller is based on a Pandaboard with an ARM CPU running at 1.2GHz, 1GB RAM and a variety of ports (HDMI video, Ethernet, digital I/O) for rapid prototyping.

8.6 Results and observations

The fabricated HOE (DCG version) is shown in Figure 80. The properties of the HOEs were measured by setting up a small collimated beam with laser diodes (532 nm, 635 nm). The collimated beam was aligned with the centre of the HOE and the output was imaged on a diffuser. The HOE tilt angle was adjusted and position of the diffuser moved until a uniform round spot was observed. The angles and focal lengths of the

fabricated HOEs for both DCG and the Photopolymer for various wavelengths are summarized in the table below:

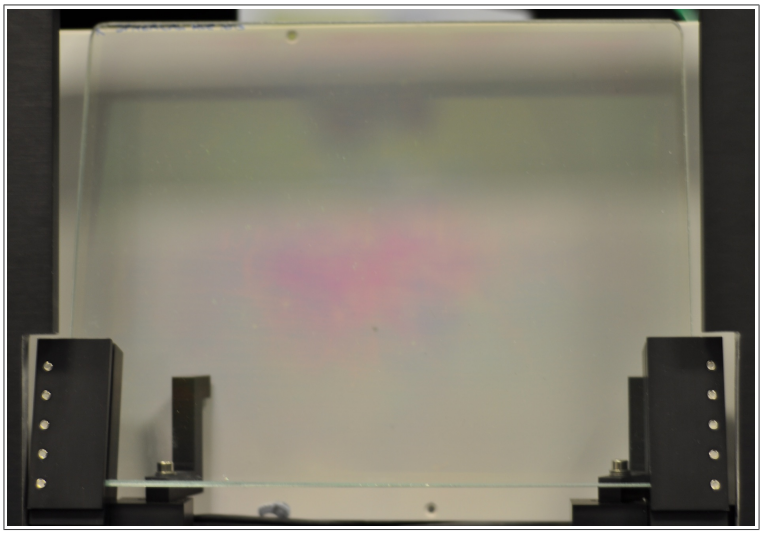

Figure 80: Holographic Optical Element

HOE Type	532nm		635nm	
	Angle (deg)	Distance (mm)	Angle (deg)	Distance (mm)
DCG	20.3	213	25.0	185
Photopolymer	19.5	204	24.3	176

Table 7: Measurements for DCG and photopolymer HOEs

The measured angles and focal distances are shown at replay wavelengths of 532 and 635 nm for both types of HOEs. The measurement error on distances is around ±1 mm. The tilt angle at 532 nm is ~20 degrees, which is in agreement with expression (86). Both HOEs exhibit similar behaviour and characteristics. The overall viewing angle of the projected holographic images is around 29 degrees, which is in agreement with expression (87). The diffraction efficiency is around 65%, for both elements, measured via the intensity of the first diffracted order relative to a straight through collimated beam. The measurement was done using a Thorlabs DET36A silicon photodetector (350 to 1100 nm range). Diffraction efficiency agrees well with expectations i.e. in between 33% (minimum for amplitude holograms) to 100% (pure phase hologram). The maximum image size before significant distortion is observed is approximately 7x7 cm.

The whole assembly is housed in a custom-built box, Figure 81. The Kinect sensor is mounted behind the HOE, which does not appear to adversely affect the infra-red performance of the device. Although a number of standard diffusers (ground glass,

plastic materials) were explored for the projection screen (to image the laser projector), it was found that ordinary non-glossy inkjet or laser printer paper works rather well.

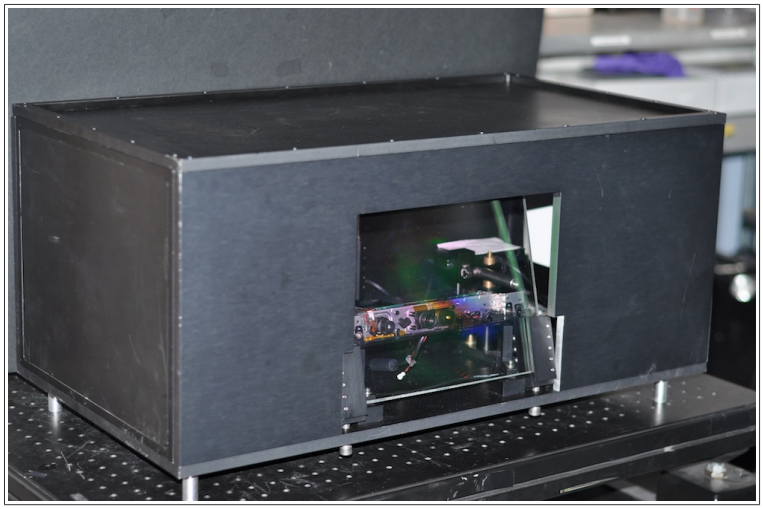

Figure 81: Interactive Holographic Display System

The display system performance characteristics to determine brightness and contrast ratio were measured with a chroma meter (Minolta CS-100). A specific test program was written to produce full-screen fully-saturated white, red, green and blue planes for brightness testing and an additional black plane for contrast measurement. The colour planes had to be measured at different distances (115 cm for green) from the display in order to fit each colour plane into the viewing objective of the chroma meter.

Given that it is not possible to produce a pure white image plane with this display, the average brightnesses for the three colour planes were measured at 6.7 cd/m^2 (red), 83.3 cd/m^2 (green) and 3.24 cd/m^2 (blue). The instrument is less sensitive to red and blue, as it mimics the human eye's response, given by the CIE colour matching function [117,118, p. 33]. The remaining analysis focuses on the green plane, where the eye is most sensitive. The luminance for green at 83.3 cd/m^2 is essentially defined by the output power of the laser projector. This value compares favourably to a computer monitor with a luminance of 100 cd/m^2, suitable for indoor viewing.

The luminance instrument did not have sufficient sensitivity to measure black, so some ambient light had to be introduced in the laboratory to get a reading, and all measurements were performed with this ambient light. The average black level was measured at .07 cd/m^2. The contrast ratio for green is therefore 1280:1, which is quite respectable. This compares well with the quoted maximum 5000:1 contrast ratio of the laser projector.

8.7 Software and Application Programming Interface (API)

The software environment for the Pandaboard is based entirely on open source approaches including the operating system and programming environment. The board runs Ubuntu Linux under XFCE4, a lightweight windowing and graphical user interface manager. All of the coding/APIs are written using the Python programming language (version 2.7) and the PyGame library (version 1.9), which is primarily designed for 2D gaming.

I wrote an API, based on advanced Pygame programming techniques [119], which handles the following core functionality:

- Icons, text, graphics
- I/O for the keyboard, mouse and Kinect sensor
- RGB screen blending
- Mirroring
- Exit and control functions

The display system maps the three RGB colours from the pico-projector in three distinct separated planes spaced according to equation (81) and are scaled differently according to (78). The red plane is smallest and closest to the HOE whereas blue is the furthest away and the largest, with green in the middle. The API is written to take advantage of these features as three different-coloured planes or screens in real space. There are specific API functions that handle the blending of the RGB screens for presentation on the laser projector. The mirroring function corrects the orientation of the projected image given the inherent inversion property of the converging HOE lens.

8.7.1 Application Code Template

I wrote a generic app template to simplify development of subsequent apps that share common features e.g. app initialisation, exit, stop, start etc. The template builds on the core API functions, dealing mainly with the layout of the screen, icons, text and graphics. The HUDApp template is boilerplate code intended to be used by other developers to simplify the implementation of their apps for the holographic display.

All of the programs share the following common keyboard control functions:

- Exit/Quit [ESC, q]
- Kinect on/off toggle [k]
- Demo mode [d]
- Printscreen/dump bitmap [p]

Kinect toggle enables or disables the motion sensor, if it is connected. The Demo mode run the app for about twenty seconds, before exiting. The Printscreen function saves the current screen and its constituent RGB planes to the disc. The files are saved in the 'bitmaps' directory and filenames are appended with an auto-incremented digit.

Most of the apps use the red and green planes with motion sensing directed to the green plane and feedback on the red plane. For example, the Keypad app (section 8.7.7) tracks the user's finger, highlighting the key icon in the green plane and shows the digit in the red plane. The red plane is also used to show control information and actions including a clear function and an exit function.

8.7.2 HUDos Top Level Menu for Apps

HUDos stands for HUD Operating System, which is a basic top-level menu, based on the template, that runs when the demo display unit boots. It shows nine tiled icons, Figure 82, corresponding to a series of apps including AirDraw, Bouncing ball, Counter, Death Star, Distance count, Keypad, Medical, Meter and Test. The apps can be launched by selection via the mouse, keyboard (through pressing the digits '1' to '9') or by the Kinect motion sensor.

Figure 82 shows the middle-right 'Keypad' tile is highlighted (inverted) in the green plane and the word 'Keypad' is written in the red plane. Note that areas of overlap between the red and green plane are yellow, as generated by the API alpha blender. The two colour planes appear in two distinct and separate planes.

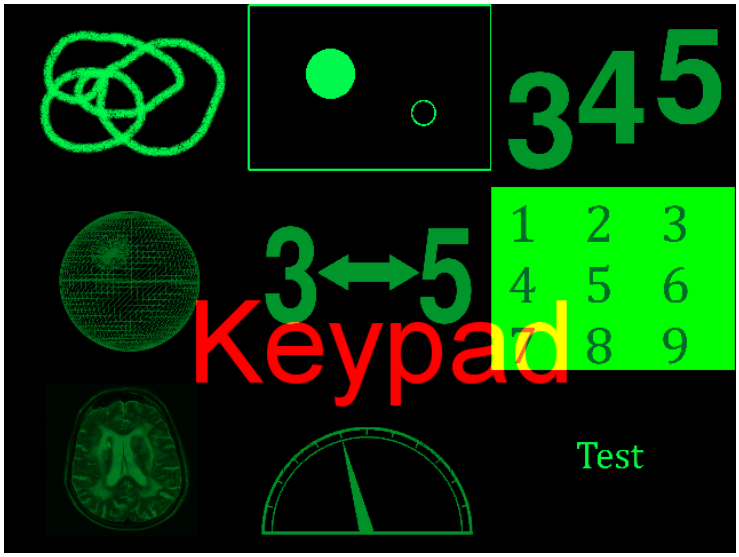

Figure 82: Top-level menu icons for app selection

The demo mode of HUDos cycles through all of the underlying apps, running each one for about twenty seconds.

8.7.3 AirDraw

AirDraw enables the user to draw simple shapes in mid-air where the motion sensor tracks a finger. The drawing is presented in the green plane. The current cursor position is represented by a green dot, which moves in response to the finger motion. The app does not respond instantaneously to the motion and there is a lag of up to a second between the movement of a finger and the dot. The app has a control plane (red) where the drawing can be cleared (or using the 'c' button) or the program stopped. The program supports mouse functionality for the drawing as well.

8.7.4 Ball

This is a simple ball game where a bouncing green circle is chased by a darker circle. The direction of the bouncing ball changes whenever the two circles collide or intersect. The darker circle is controlled by the user with the mouse or Kinect motion controller.

8.7.5 Counter

The Counter app is a simple counter that shows the digits zero to nine in any of the three different planes, depending on the colour. The separation of the three different colour planes can be clearly seen. The digits simply count up in one second intervals. The colour planes can be switched by pressing the space bar or movement via the motion sensor.

8.7.6 Deathstar and Medical

These apps share the same codebase, *animateImages*, which loads and replays a pre-determined sequence of still images at regular intervals to perform some simple animation. One hundred animated Deathstar images were generated from a 3D Model using the modelling program 3D Studio Max. The medical images are a set of twelve CT scans downloaded from the internet and converted from greyscale to green using Matlab. The user has an option to stop the animation ['s' key] and sequence the stills manually using the space bar.

8.7.7 Keypad

Keypad tracks the user on a keypad whilst highlighting each selected key in the green plane and displaying the corresponding digit in the red plane. The tracking is performed on the mouse or the motion sensor. This app enables the user to 'touch' the icons that appear in mid-air as well as activate events when the icon is activated or de-activated.

8.7.8 Meter

This is a simple app that displays a meter with six steps or segments with a dial

indicator. The position of the dial depends on the average depth detected by the motion sensor across nine zones.

8.8 Acknowledgements

This work was part-funded by the ESPSRC, the UK Engineering (Physical Sciences Research Council), and the TSB, Technology Strategy Board. I gratefully acknowledge Dinesh and Joy Padiyar from TripleTake Holographics for their expertise and skills in fabricating several versions of the HOE in DCG and photopolymer. I would also thank Dr Thomas Faecke from Bayer Material Science for providing access to their early experimental photopolymer material.

Although most of the apps and APIs were written by myself, the Kinect interfacing software layer and integration into the apps were written by Chi Can, who also conducted the diffraction efficiency measurements under my supervision. She also wrote the following apps based on my APIs and templates: AirDraw, Bouncing Ball, Distance and Meter. The Deathstar images were modelled and rendered by David Viallard, a 3D graphics designer.

8.9 Conclusions and further work

The work shows that HOE-based approaches to holographic displays are a viable and practical approach that is readily implementable with largely off-the-shelf components. There is good general agreement between the theory and implemented HOEs in terms of focal distance (20 cm in green), depth of field across the colour image planes and replay angles (29 degrees). The display achieved good performance with bright images that are clear and visible under ambient laboratory lighting conditions (83.3 cd/m^2). Diffraction efficiency is also reasonably good.

Barrel distortion arises when the paraxial imaging condition is violated [120,121], where the object size should be constrained to small dimensions along the optical axis. A large aperture lens enables bigger images to be reconstructed; so increasing the size of the aperture would decrease the level of distortion. Further work would be required to characterise the uniformity of the image and optical distortions arising. This would require more sophisticated image acquisition and processing algorithms. Once characterised, the information may be used to correct for distortions by embedding a corrective image processing layer within the API.

The resolution of an optical system is given by the Rayleigh criterion, which describes the ability of the system to resolve a pair of identical adjacently-placed incoherent point sources. The Rayleigh criterion [122, p. 59] states that the resolution of the system is

defined by the distance between the peak diffraction pattern of one source and the trough (first minimum) of the second source. In this case, the system is said to be diffraction limited and the resolution is given by the following equation:

$$\Delta = 1.22\lambda\frac{f_0}{H} = 1.22\lambda f_{\#} \tag{88}$$

Where H is the height of the HOE. Given that our system has an *f/#* of unity, the resolution of the system at 532 nm is Δ=649 nm; this is several orders of magnitude smaller than the resolution of the laser projection system (only WXGA pixel resolution). Hence there is ample headroom for improving the resolution of the HOE display in future, by simply using a better laser projector (e.g. HD resolution). Since the laser projection system is coherent, the Rayleigh criterion cannot strictly be applied [58, p. 157] as the minimum separation is dependent on the relative phase between the sources. However, the laser projector contains optics and electronics to reduce speckle and the image is formed on a diffusion screen; all factors that serve to make the light semi-coherent. It follows from equation (88) that decreasing the wavelength or *f/#* would also act to improve the resolution of the system.

The work presented here was a HOE made with laser interference in a photosensitive material. It is possible to fabricate such a component as a DOE (sections 1.4 and 4.2). Such a structure is known as a Fresnel Zone Plate for an amplitude-only component and a Fresnel Phase Plate (FPP) for a more efficient phase-only version [123, p. 6]. The FPP is amenable to low-cost mass production via embossing (section 1.6.3). However, the challenge is to make a large-area master via photolithographic or other means.

The display system achieves near real-time performance despite the apparent limitations of running an interpreted language. For example, in the AirDraw app, the lag between movement of the finger and the actual drawing action is around 2 seconds. This is noticeably sluggish but acceptable for an early-stage demo. Future motion controllers are expected to have faster real-time behaviour. To improve software performance, it is possible to run the hardware without the graphical interface directly from the console since both Python and Pygame support non-GUI operation. Further optimizations could be gained through coding parts or the entire software directly in C/C++; although this would affect the productivity and turnaround time for new applications.

The APIs and templates enable fast code turnaround, which is very useful for rapid prototyping of new applications. A suite of ten programs was coded in just a few weeks.

The programming environment enables apps to be developed on any computer, since Python and Pygame have cross-platform support. Indeed, many of the apps were developed on a Mac without a Kinect and the same code base runs on the Pandaboard with a Kinect without any modifications. In this case, the Kinect functionality is modelled using the keyboard and mouse.

The work was presented at the 2013 SPIE Photonics West Conference and it is published in the proceedings [124]. The display appears to be generally well-received by the holography community and the public at large. The display won the IHMA 2013 Best HOE Application award. It has appeared on UK national television, on Channel 5's 'The Gadget Show', with over a million viewers, broadcast on 25th March 2013 (section 9.3). We were also invited to their Live Event exhibition, held between 2-7th April 2013; where we received an estimated 15,000 visitors, many of whom specifically came to see the HOE display following the TV broadcast. This underlines the fact that people really expect and want a holographic display with floating images in mid-air with interactive capabilities. Indeed, the work shows that such displays are no longer within the realm of science fiction but are finally an engineering reality.

Chapter 9 - Conclusions

The work conducted within this Engineering Doctorate thesis has led to a fresh approach for realising real-space holographic displays. The work sets the foundations for an enabling technology that opens up new avenues of research, which will ultimately lead to the ideal holographic display portrayed in the popular media. All of the work presented in this thesis is repeatable at relatively low cost with good quality, large images that are bright and visible under ambient indoor lighting conditions.

The work presented in this thesis demonstrates how to realise 3D images that are substantially larger than conventional approaches, and thus easier for humans to interact with. This has been achieved by exploiting the scope that exists for optimisation through compression and spatial multiplexing of the information in the hologram. These properties are explored and exploited throughout the thesis. The work demonstrates that fabrication of the holographic screens is relatively straightforward. I have shown that my approach is viable with a variety of standard and advanced holographic techniques, including both transmission and reflection holograms using:

- Classical interference holography with physical masking and reconstruction via laser projectors (Chapter 2)
- Digital reflection holograms on holoprinters with software masking and reconstruction via an LCD screen (Chapter 3)
- Computer Generated Holograms and SLMs using Fourier Optics simulation models and SLMs for demonstration (Chapters 4 to 7)
- Holographic Optical Elements using Fourier Imaging and a converging diffractive lens (Chapter 8)

Transmission holograms are best suited to large holographic displays as demonstrated in Chapters 2, 7 and 8. This is fortunate because transmissions are easier to fabricate, requiring simpler materials with lower resolution than reflection holograms. The images are good quality in terms of image resolution, brightness and clarity. The images are also surprisingly speckle-free given that transmissions require laser illumination. This is partially down to luck (modern laser projectors include anti-speckle mechanisms) and use of diffusers in recording or playback. However, not all of the approaches explored in the thesis were ideal and problems were encountered with the holoprinter (Chapter 3) and the DOE fabrication (Chapter 4). These were largely due to parameters outside my control, relying on third party suppliers or equipment. Nevertheless, there is plenty of

scope for improvements and future work, as highlighted in the respective chapters. Given the wide array of holographic fabrication methods explored here, I am confident that the techniques developed in this thesis are directly applicable to other types of holograms including rainbow holograms [125,126], edge-lit holograms [127,128] or other exotic hybrid holographic techniques [129,130].

In the spirit of an engineering doctorate, the research in this thesis has been driven by practical considerations towards proof of concepts and prototypes with commercial constraints in mind, rather than pure scientific discovery. The holographic displays are largely realised with conventional off-the-shelf components based on standard processes with some modifications. I have shown that the bottom-up technical approach to making simple holographic displays is viable and feasible with today's technology. Furthermore, the approach is scalable through leveraging general improvements in technology evolution in photolithography, micro-optics, lasers, projectors, embedded computers, GPUs etc. Embedded computing hardware was based on Intel-compatible processors at the start of the project with poor I/O support under Linux. During the project, ARM-based boards were introduced at low cost, low power and smaller size with excellent I/O support. The same period has witnessed a tremendous growth and popularity of GPU technology for high performance computing.

The displays have progressively improved with each iteration through the chapters. The project has met most of the objective initially set out in section 1.7.1. Indeed, the project went significantly beyond the original objectives with robust code, interactivity and opening up the possibility of high resolution displays. This has been a multi-disciplinary project that pulls together elements of theory, optics, electronics, manufacturing and software engineering. The following list outlines the main areas of activity:

- Holography: interference holography, digital holograms, CGH, DOEs, HOEs
- Optics: geometric optics, Fourier Optics, Fourier Imaging
- Electronics: digital hardware, analogue hardware, embedded systems
- Optoelectronics: diode lasers, laser projectors, motion sensors
- Signal and image processing: modelling, simulation and supercomputing (GPU)
- Software: high-level applications, control code, APIs

The work has taught me the value of basic theory, numerical simulation, modelling and emulation in the absence of funds to fabricate key components. I feel that the models and simulations provide a solid foundation for the work going forwards.

The work has entailed a great deal of software development in the form of algorithms, control code, application code, image and signal processing to name just a few. I have written over 10,000 lines of code in this project, although individual functions are just a few hundred lines. All of these have culminated in a series of tools like FOURIOT for design space exploration. In this case, the tool as a whole is greater than the sum of the constituent parts. FOURIOT will be released in the public domain and I hope that it will be useful in the teaching and understanding of Fourier Optics and imaging.

In terms of software performance, high level scripting and interpreted languages such as Matlab and Python are used throughout this thesis. Python in particular has been instrumental in achieving a great deal of functionality with a high degree of productivity by minimising the amount of coding with quick turnarounds. The code is written to be robust, readable, modular and easy to upgrade or expand. A key feature is portability which means that the same code can be run on any computer including Apple Mac, Unix or Windows operating systems running on any type of processor (AMD, Intel or ARM). The performance penalty of using scripted languages is not only offset by the general improvements to computers following Moore's Law, but also in general performance increases in the language, APIs and libraries. Indeed, the performance of Matlab code can be increased through the use of an additional parallel computing toolbox to leverage multiple cores plus GPUs; additional gains are achievable via compilation into an executable.

All of the display software runs on the free and open source Linux operating system and Python. Matlab, on the other hand, is an expensive commercial language and tool; although open-source equivalents such as Octave exist but this has not been explored. Given that Python has numeric packages such as Numpy and SciPy [131], which offer Matlab-like functionality; I do not envisage great problems in porting the Matlab code to Python. Extensions to support GPU programming are now emerging.

9.1 Recent research

This section highlights some recent relevant research in the field of holography or related optics and discusses ways in which these could be leveraged towards realising holographic displays.

The fabrication of controllable Bragg layer structures has only recently become possible via interferometric modulator display technology [132,133]. The modulator is based on cell elements comprising a thin film stack acting as a Bragg layer together with a membrane electrode. The cell thus reflects a single wavelength of ambient light.

When a voltage is applied, electrostatic forces bring the membrane into contact with the stack, causing the Bragg effect to collapse such that the cell acts as an absorber. The advantage is that the technology only requires ambient incoherent light, just like a reflection hologram. This technology is currently used for colour e-book reader displays that do not require a backlight. In principle, this technology could be extended to synthesise digital reflection holograms to perform arbitrary Bragg diffraction and spatial light modulation. One way of performing this could be to merge this technology with Digital Micro-mirror MEMs in order to get angular reflectivity; however, this appears to beyond the current state of the art.

HP Labs are working on spatial multiplexing of fixed grating structures [134,135] to achieve beam-steering of an LED backlight to produce wide-angled 3D images, Figure 83. In some ways, this use of spatial multiplexing validates the approaches taken in this thesis, although this is done by HP at a much finer resolution, somewhere between that of a hogel/holopixel of a digital hologram (section 1.3.3 and chapter 2) and a pure-phase SLM. Both of these approaches would still suffer from the scalability issues associated with a large area display (section 1.2.5) in terms of computation and bandwidth. There are still some considerable fabrication engineering issues to be overcome before practical displays could be realised.

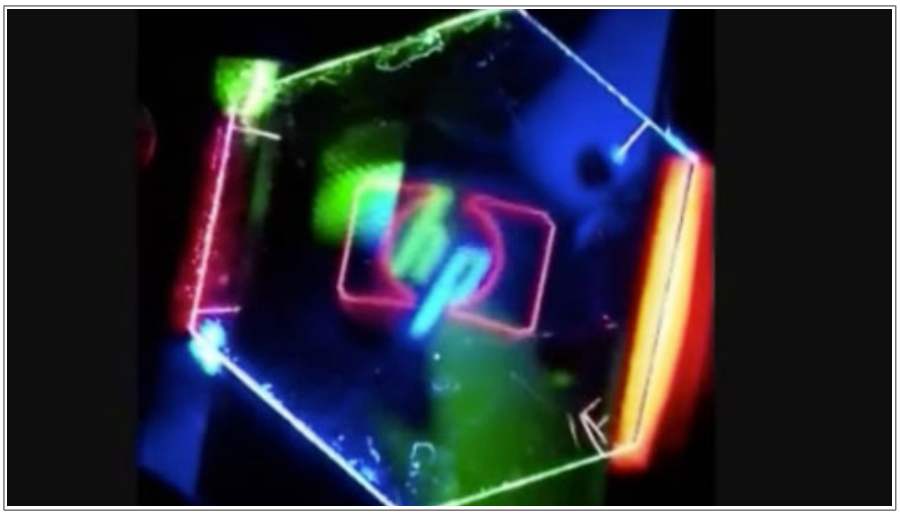

Figure 83: HP Labs grating display, from [134]

MIT has recently developed a tuneable laser phased array [136] that could be used for steering light in any direction for holographic projection instead of SLMs. They have also developed a hybrid acousto-optic modulator coupled with a waveguide that can be used for beam-steering light propagating through the waveguide [137]. The advantage is that multiple wavelengths can be steered simultaneously and the core technology is

potentially low-cost but the steering electronics are far from trivial.

Further exotic approaches to a HOE/DOE could be around negative refractive index meta-materials that can behave as flat lenses [138,139] operating all the way from microwaves to visible light and UV. The meta-material is formed from stacked layers of nano-scale material of negative electrical permeability (ε) and negative magnetic permeability (μ). In this case, the stack is formed from alternating layers of silver followed by titanium oxide, creating a refractive index $n=-1.0$, where point sources of light from one side of the lens are focussed on the other side at the same distance. This leads to a 3D mirror image of a translucent object with unity magnification; whereby a point at (x,y,z) is imaged at $(x,y,-z)$ with $z=0$ as the plane of the lens [140]. Unlike the HOE/DOE lenses, the images created are aberration-free and the lenses can be made in large-area formats with no optical axis. It is also possible to make these lenses optically active using shorter wavelengths as a switching mechanism, enabling use as a shutter or potentially a variable-focal length lens.

The methods reported within this thesis could be enhanced by any of the above developments. Indeed, an objective within the programme was to develop approaches that can exploit, rather than be rendered redundant by, such technology developments.

9.2 The near future

The work done throughout this thesis has plenty of scope for further research and enhancements to the techniques developed. This can be seen from the concluding sections of most chapters. The extent to which further work may be conducted ultimately depends on the commercial interests in future products as well as budgetary constraints.

One of the disadvantages of the optical architecture of the HOE display from Chapter 8 is the overall mechanical bulk of the system. The volume of this bulk is proportional to the focal length of the HOE i.e. $\sim 2f_0$, so there is an interest in reducing f_0 but this is not practical with the current holographic optical recording setup, which can only achieve f/# of unity. However, this constraint no longer applies if the HOE is fabricated photolithography as a DOE, where f/# of less than unity are possible. The challenge here is to find a fabrication facility that can handle large-area DOEs.

The HOE display from Chapter 8 can be viewed as a set of discrete 2D planes imaged in real 3D space via a wavelength-dependent lens. The next logical step is to place many more 2D planes adjacently to form a 3D volume. This can be achieved in several ways. The previous chapter highlighted the wavelength dependence of the HOE focal point,

leading to three distinctly coloured (RGB) planes in space. Hence, if it is possible to alter the wavelength of the projection subsystem, then it would be possible to build a multi-planar volumetric holographic display. This technical approach is being taken for the next generation display. We have won a TSB grant to implement this and it is being designed for the presentation of medical images from a CT scanner. Other interesting approaches to making a volume holographic display based on the HOE architecture include the prospect of an active DOE, which can sweep the volume electro-optically using tuneable diffractive liquid crystal lenses [141,142].

9.3 Project dissemination and outputs

An overview of all outputs generated from the start of the project is presented in Table 8 below. This not only includes scientific papers but also articles in the trade, engineering and popular media (including newspapers and television). The key technical outputs include one granted patent [76] in the USA with other territories to follow; plus two published SPIE papers [75,124].

Types of output:
J = journal article
C = conference proceeding
O = oral conference paper
OI = oral conference paper – invited talk
P = poster conference paper
A = press article (possibly written by a journalist)
I = internal company report
X = other

Year	Type	Full reference
2009	P	J. Khan, EngD Conference, “Low Resolution 3D Holographic Displays”, Jul 2009
2009	X	Patent, J. Khan, “A Three Dimensional Holographic Volumetric Display,” PCT Patent No. WO/2009/109785, International.
2010	P	J. Khan, “Holographic Volumetric 3D Displays” presented at the Institute of Integrated Systems Workshop, Edinburgh, Jan 2010
2010	O	J. Khan, I. Underwood, “A pico-projector driven 3D Holographic Volumetric Display” presented at The SID Personal Projection and Information Displays Conference, Dresden, Mar 2010
2010	C	J. Khan, I. Underwood, A. Greenaway, and M. Halonen, “A low-resolution 3D holographic volumetric display,” in *Optics, Photonics, and Digital Technologies for Multimedia Applications* **7723**, P. Schelkens, T. Ebrahimi, G. Cristobal, F. Truchetet, and P. Saarikko, Eds., p. 77231B–7, SPIE, Brussels, Belgium (2010).
2011	A	I. Lancaster, “Holoxica’s Dynamic Hologram Display” in *Holography News* **25**, pp. 1–2 (2011).
2011	O	J. Khan, EngD Conference, “Animated Holographic 3D Displays”, Jul 2011
2011	OI	J. Khan, ‘3D Holographic Visualisation’ presented at The Photonex Conference, Coventry, Oct 2011
2011	OI	J. Khan, ‘Animated Holographic 3D Displays’, presented at The HoloPrint-HoloPack IHMA Conference, Las Vegas, Nov 2011
2012	A	D. Jeff, “The 3D Force may be with Holoxica”, Scotland on Sunday newspaper, Jan 2012
2012	A	C. Chinnock, “Holoxica Debuts Dynamic Hologram Display,” in *Large Display Report* **10**, pp. 42–43 (2012).
2012	OI	J. Khan, ‘Fixed Frame 3D Holographic Displays’, presented at SPIE Photonics West Conference, San Francisco, Holography Group Meeting Jan 2012
2012	O	J. Khan, K. Nguyen, I. Underwood, “New Applications of Holographic Projection in 3D Image Capture and Display”, presented at 4th Workshop on Liquid Crystals for Photonics 2012, 9-11 December, 2012, Hong Kong
2012	OC	J. Khan, C. Can, A. Greenaway, and I. Underwood, “A real-space interactive holographic display based on a large-aperture HOE,” in *Proc Spie* **8644**, p. 86440M–86440M, SPIE, San Francisco (2013) [doi:10.1117/12.2021633].

2013	A	C. Chinnock, “Holoxica Demonstrates Second Generation HOE-based 3D Display”, 3D News Report, Mar 2013
2013	AX	TV Broadcast, Channel 5, ‘The Gadget Show’ featuring HUD-style Holographic Display, Jason Bradbury, 25th Mar 2013
2013	A	I. Lancaster, “Real-time ‘Thin Air’ Display from Holoxica ” in *Holography News* **27**, pp. 1,4 (2013).
2013	P	J. Khan, “Interactive Holographic HUD-style Display”, SID Mid-Europe Conference, Apr 2013, published in proceedings
2013	O	J. Khan, “3D Holographic Imaging of Scientific, Molecular and Medical Datasets”, presented at Sinapse Annual Scientific Meeting, Aberdeen, May 2013
2013	A	J. Khan, ‘Biomedical holographic 3D imaging’, Laser Focus World, Jul 2013

Table 8: Project dissemination and outputs

9.4 Business dimension

Holoxica Ltd and this project share a substantial business component that warrants some explanation that is atypical for many Engineering Doctorate programmes. The majority of Engineering Doctorates are funded by relatively large or medium-sized organisations. This project can be considered to be at the extreme end of the spectrum, where the company was formed from scratch with the research providing the basis for the company's future development. As such, the project faced additional commercial and resource risks that established organisations take for granted. For example, this includes substantial business planning, fund raising, writing grant applications, networking, project management and so on.

The taught component of the Photonics EngD programme includes five modules from the MBA programme. These courses were invaluable for starting up the company in terms of administration, financial planning, accounting and business/strategic planning. This is a genuine story of a high tech startup formed from an idea in a basement, all the way to several generations of proof-of-concept demonstrators.

The company raised about £200k from private shareholders and an additional £200k from grants in order to perform or support much of the work done in this thesis. The work done here provides a solid foundation for the company going forwards in terms of attracting additional investment and commercialisation of the technology into products.

9.5 Acknowledgements

The contributions of various organisations, individuals and others are acknowledged throughout this thesis. The Photonics Industrial Doctorate Centre deserves credit for taking on such a risky project and working outside their normal comfort zone with a startup. Holoxica's shareholders are hereby acknowledged for sharing the vision of a true 3D holographic display. The project has benefited from the following grants and awards, all of which have contributed in one way or another:

- EPSRC grant GR/T11289/01 for the Engineering Doctorate studentship
- Scottish Funding Council, TTOM Award #C5708
- Technology Strategy Board feasibility grants 130559 and 130893
- Technology Strategy Board Proof of Concept grant 710253
- Scottish Enterprise, TTOM Award

In addition, the following prizes indicate industrial relevance and potential for the future of the technology:

- Santander Bank Edinburgh Entrepreneurs Pitching Contest, Mar 2010
- Thales Scottish Technology Prize, Oct 2010
- SPIE Scholarship Award, Jan 2012
- IHMA Best HOE Application Award, Nov 2013
- AEngD Engineering Writer of the Year Award, Nov 2013

These awarding organisations are leading bodies in photonics and engineering. I am very grateful for their insights and recognition of this work.

9.6 Conclusion

Stereoscopic 3D TV and cinema have peaked during the lifetime of this thesis and these markets are now in decline, because of low levels of satisfaction with current stereo 3D. This was entirely predictable and is an example of history repeating itself. I have been fortunate to have received extensive press and media coverage throughout this period, which highlights the still-unsatisfied demand for a true 3D display. I aim to address these demands in the coming years. Indeed, there are relatively few 3D/photonics-related doctoral theses that can show this level of public dissemination and interest [30,32,143,144].

The Engineering Doctorate has allowed a small startup company with severely limited resources to achieve a great deal, starting from just an idea to holographic displays that are practical, scalable and commercially viable. There is a wide gap between the ideal

science-fiction holographic display and current volumetric displays (section 1.2.3), which is bridged by the research presented here. The spatial multiplexing and HOE solutions proposed in this thesis are simple, elegant, practical and scalable. The work presented in this thesis is rooted in solid engineering reality, rather than science-fiction fantasy, which means that practical holographic displays can be realised today.

Our early results compare favourably with leading holographic display research by major organisations such as Zebra Imaging (section 1.2.5) in terms of some basic parameters (brightness, contrast ratio, frame rate, bulk, power consumption). This is quite an achievement given Holoxica's modest budget. Zebra Imaging has received over $25M from DARPA to realise their prototype. Our optical architecture is simpler and lower cost without requiring exotic photonic and optoelectronics components. Similar arguments apply to comparisons with holographic displays by other competitors such as SeeReal (section 1.2.5).

Mainstream research in holographic displays has tended to focus on computation in the electronic domain, drawing upon digital Fourier Optical techniques. However, this approach is very problematic not only from a computation and bandwidth perspective, but also in terms of practical implementations that are beyond current fabrication technologies. The work in this thesis challenges the prevailing view by shifting the emphasis back into the analogue electro-optical domain via spatial multiplexing holograms or Fourier imaging with holographic optical elements. It turns out these methods are not only feasible with current technology but are also highly scalable with big, bright and clear images. The proposed solutions can deliver almost immediately at a reasonable cost. The penalties for all this simplicity are constraints on the reconstructed image in terms of voxel resolution, opacity and lack of occlusion. However, this is a price worth paying for a solution that works and is good enough for a wide range of applications in medical imaging, security imaging, scientific visualisation and engineering design.

The work in this thesis is just the starting point for an entirely new area of research, which has largely been overlooked by the mainstream holographic display community until now. As the technology is refined with further research, it will be possible to introduce colour mixing, voxel directionality and other features to ultimately yield the ideal holographic three-dimensional display.

Appendix A – Image and Hologram Mask Patterns

This appendix depicts the mask patterns used to create the source images as well as the masking of the holographic plate for recording. The use of these mask patterns to create the holographic screens are described in detail in Chapter 2. The graphics in this appendix are not to scale and in some instances have been inverted for convenient representation on print. In addition to digits, segments and grid patterns, there are a number of markers comprising crosshairs and circles: these are used for alignment between successive recording steps.

All of the graphics are 200x300 mm. They are designed and meant to be viewed in a landscape orientation. The font type is Arial. Only two of the four grid masks for 2x2 holoscreens (for numeric displays) are presented in Figure 87 and Figure 88; depicting the mask patterns along two rows (as viewed in landscape format). Only one of the nine grid masks for the 3x3 holoscreen (for segmented displays) is presented in Figure 90. Only a limited subset of the total number of grid masks are presented here because they are fairly similar, the differences being spatial displacement in the horizontal or vertical directions.

Please contact the author if originals are required.

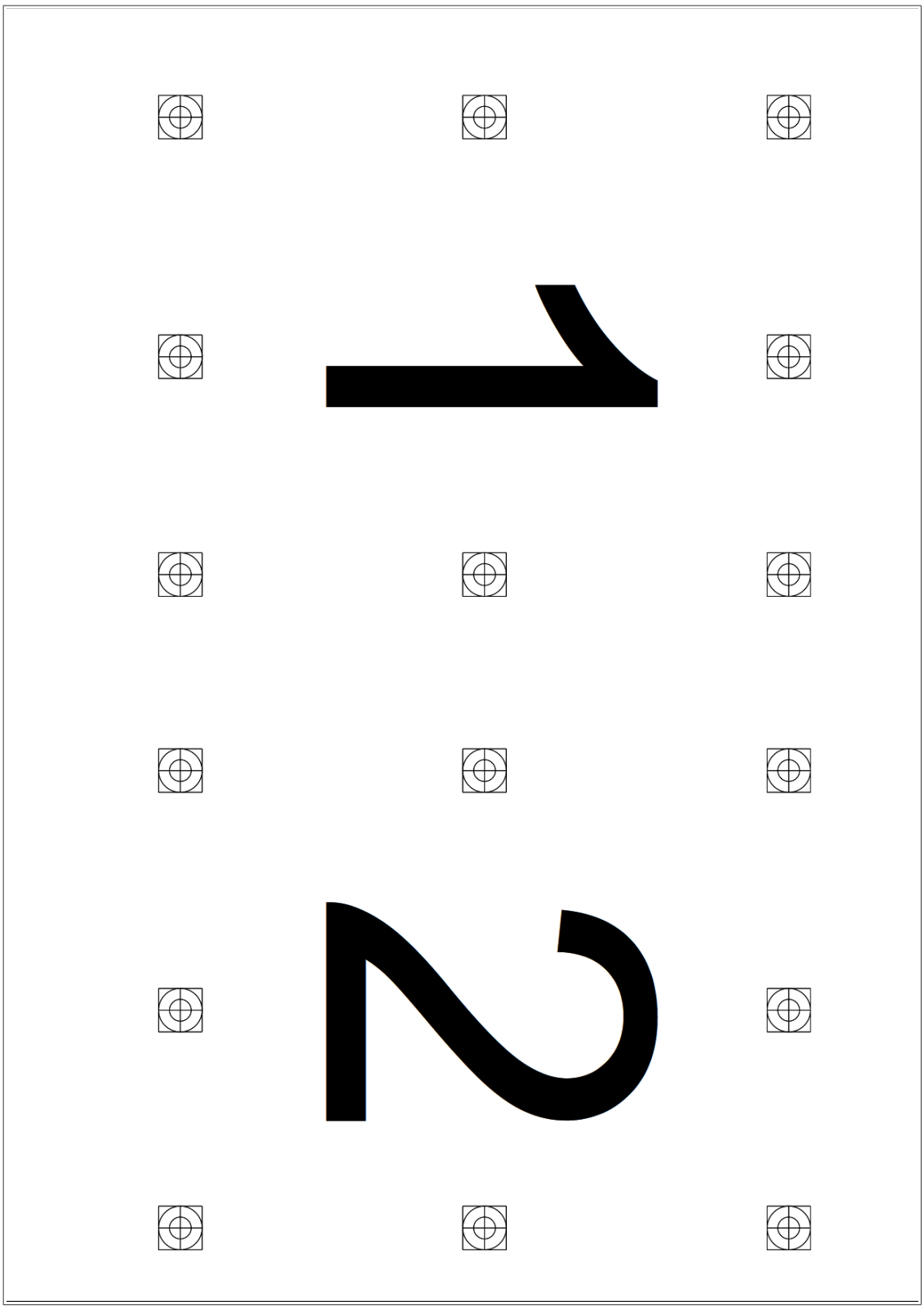

Figure 84: Digit images for overlapping display (inverted, 1 of 2)

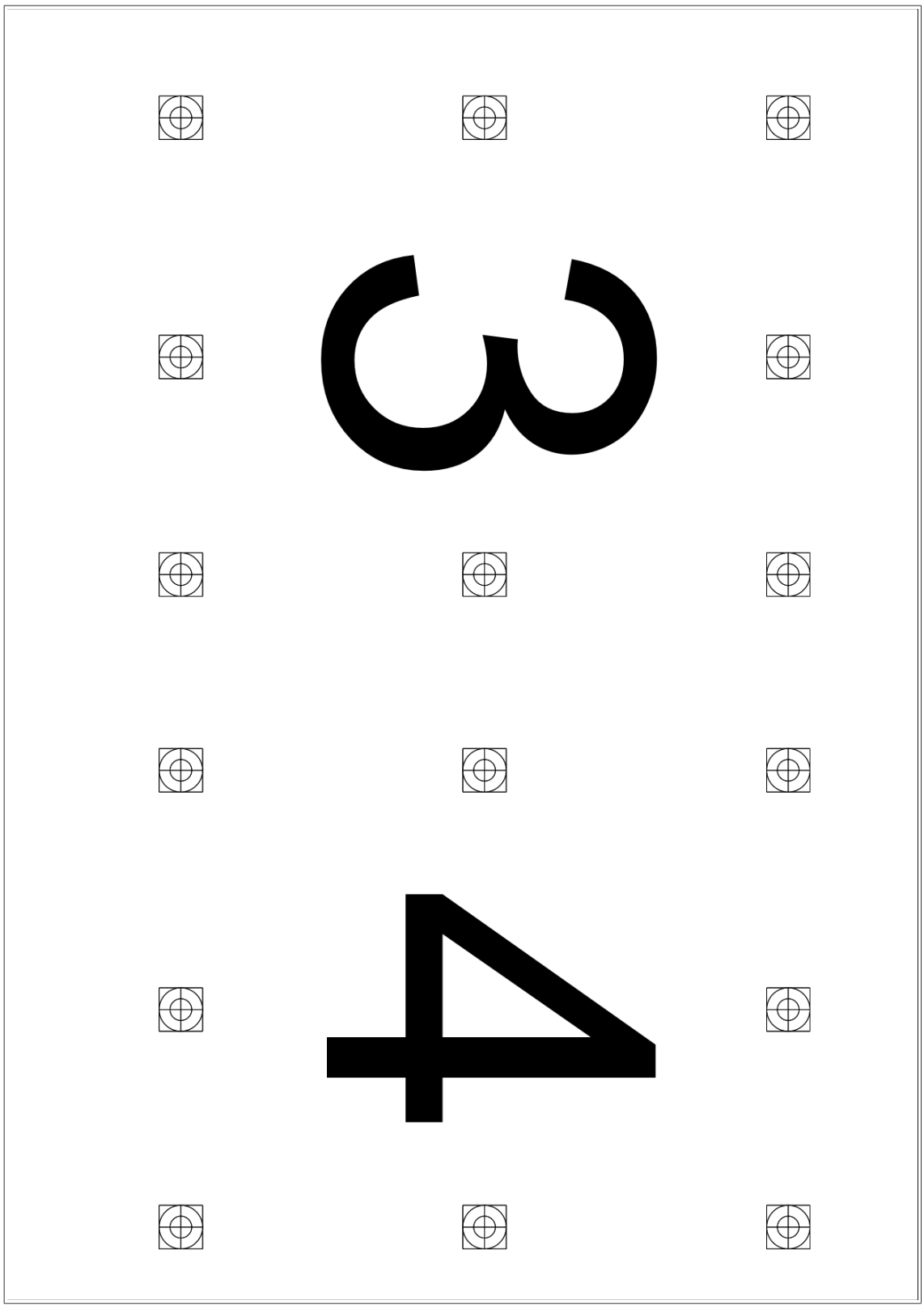

Figure 85: Digit images for overlapping display (inverted, 2 of 2)

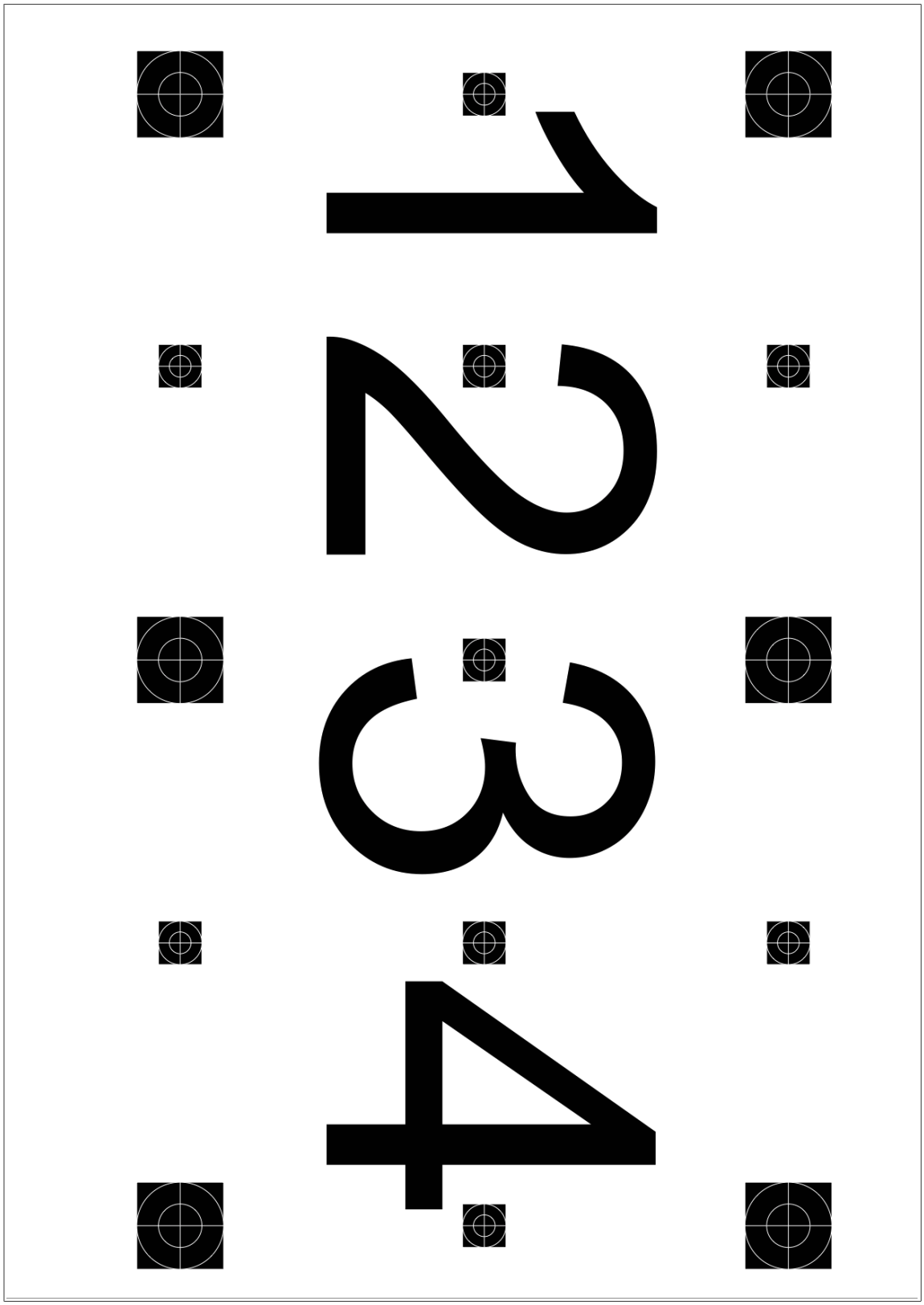

Figure 86: Digit images for side-by-side numeric display (inverted)

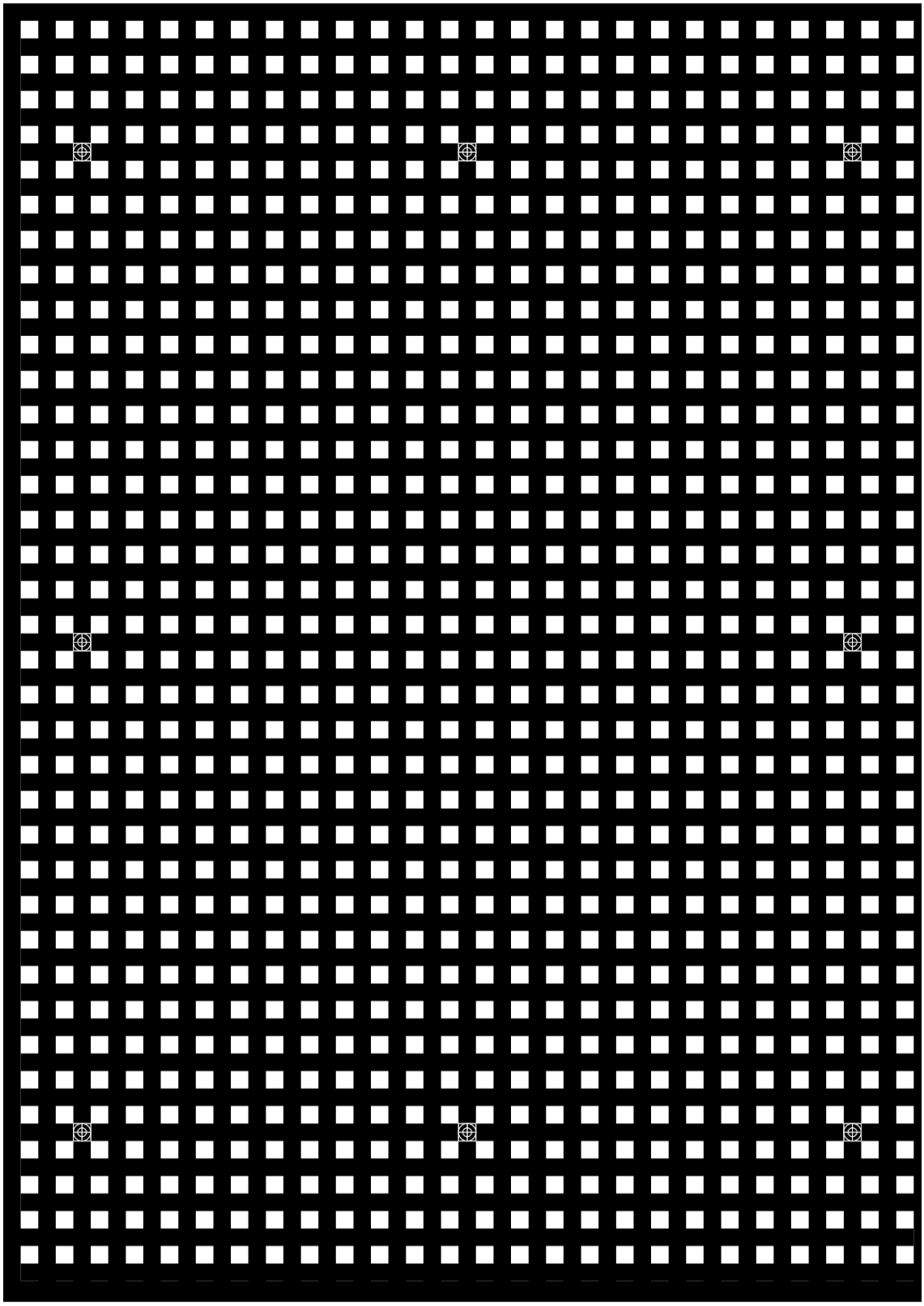

Figure 87: 2x2 Grid Mask for hologram recording (4mm): 1 of 4

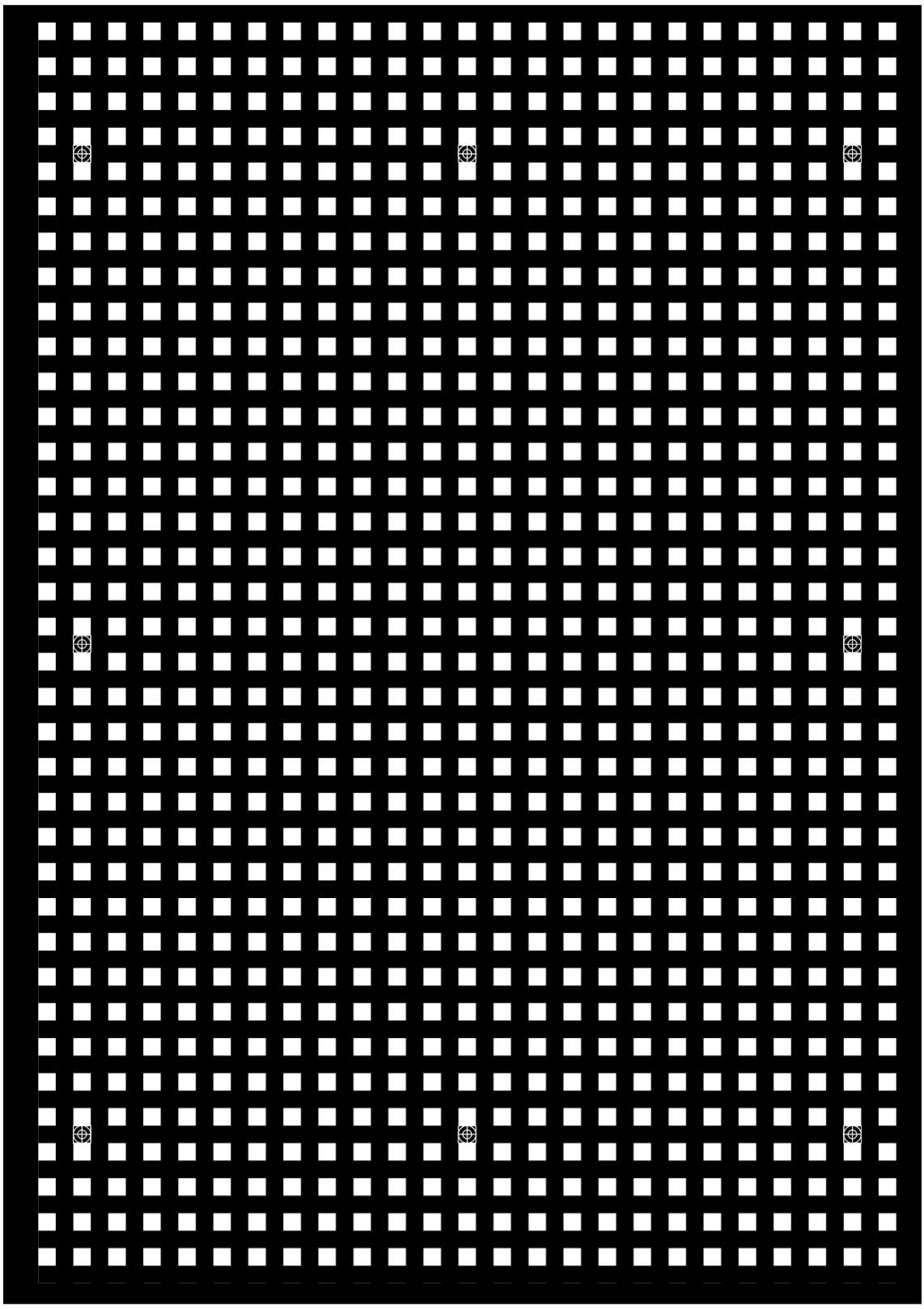

Figure 88: 2x2 Grid Mask for hologram recording (4mm): 2 of 4

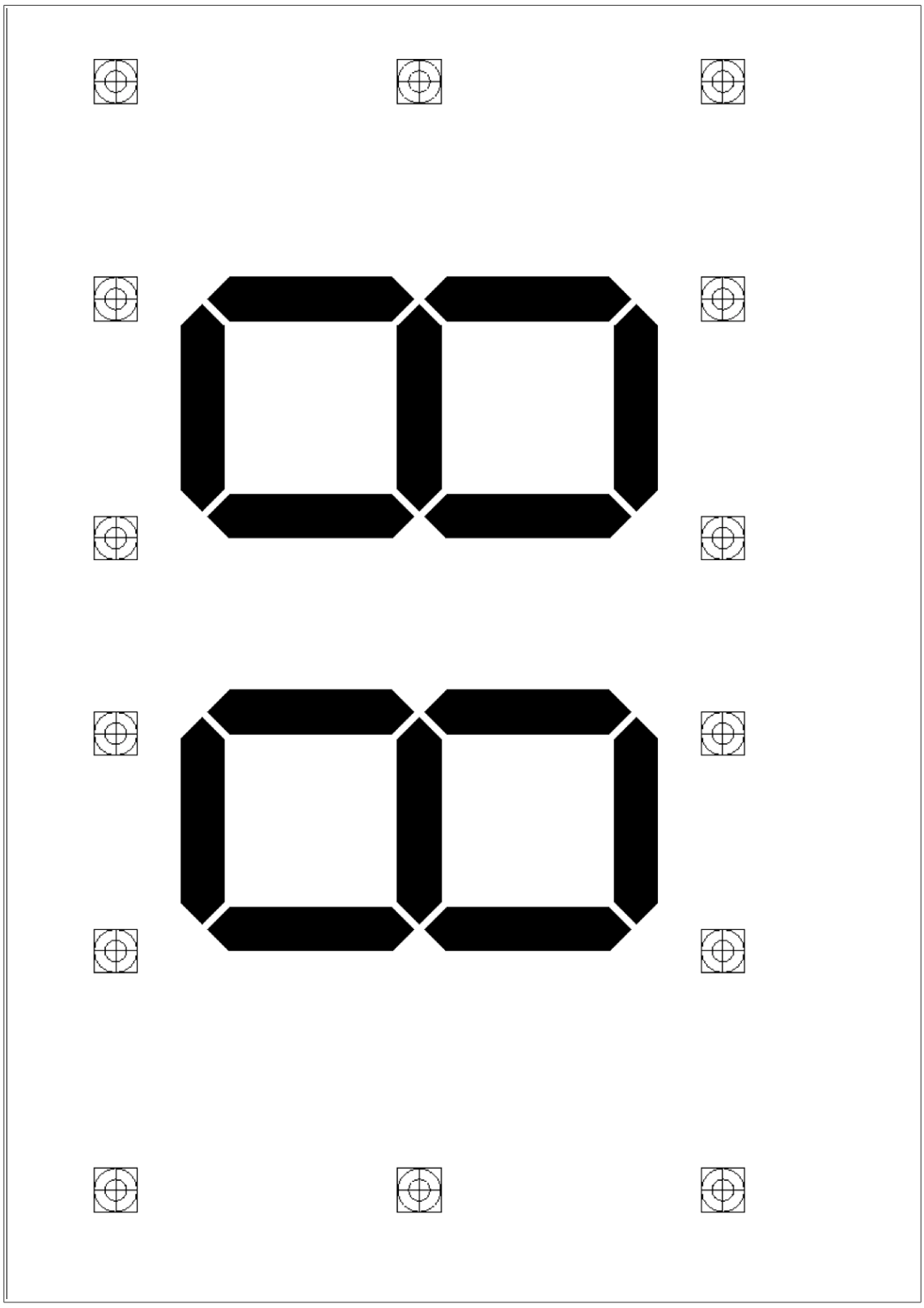

Figure 89: Twin seven-segment style display (inverted)

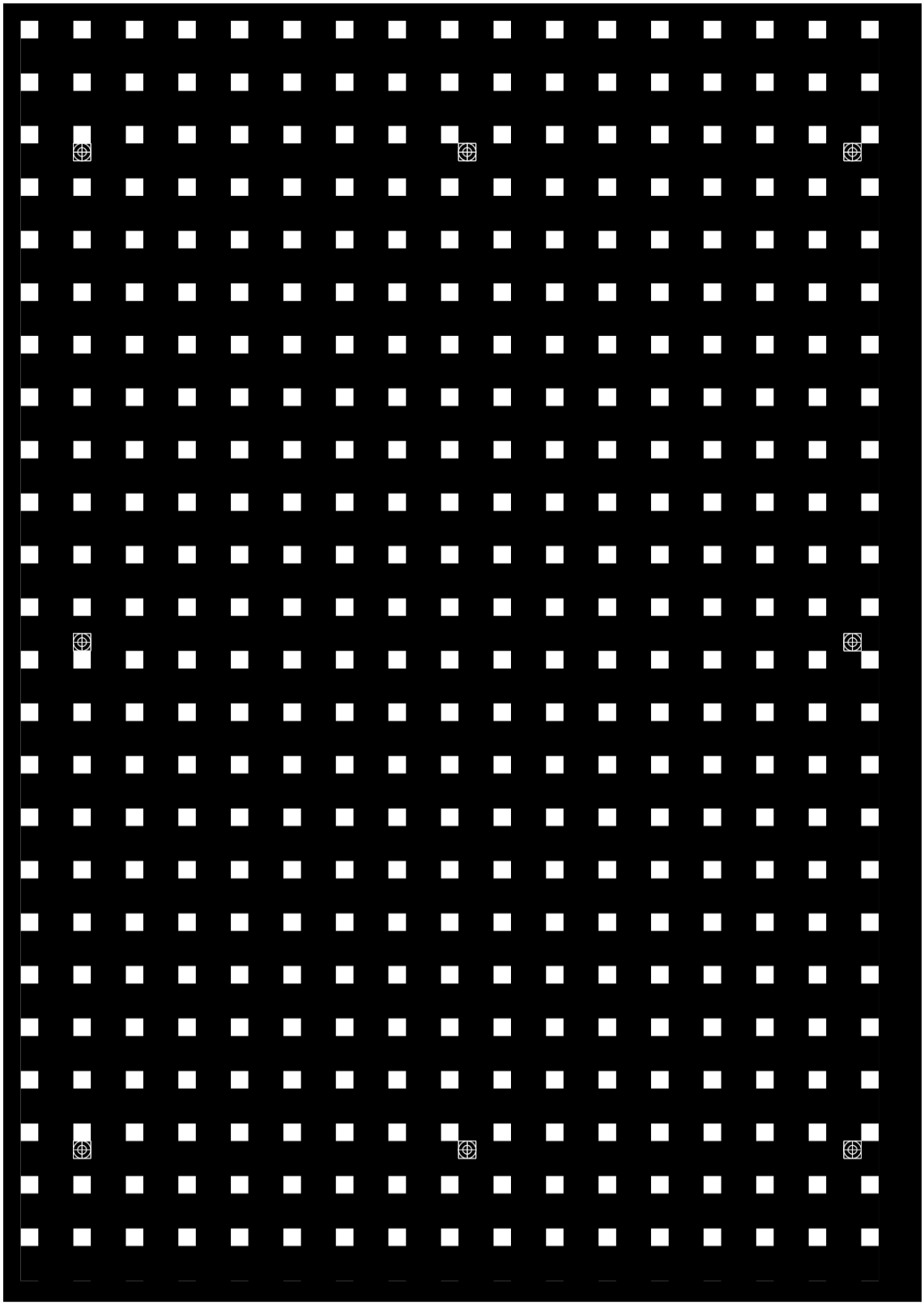

Figure 90: 3x3 Grid Mask for hologram recording: 1 of 9

Appendix B – Python Pattern Generation Code

This appendix presents an outline of the Python code used to generate the patterns and sequences to drive the laser projector or LCD to mask the sampled holographic screens.

The code file name is: testpatterns.py

This is an early release version 1.5, which is stand-alone and stable. This is a reasonably compact version with only ~200 lines of code, representing the top-level code with the core functionality. This code uses two further library classes:

- config.py to read configuration files (pattern definitions and sequences)
- patterns.py to represent tile and pattern geometries

The code for these programs is not presented here and all three programs contain about 460 lines of code, excluding comments.

The current version is 3.1, with a host of further features including network control and bitmaps. This code is over 1,000 lines, more than double the size of version 1.5.

Further details of the operation of this code is presented in section 2.3.3.

```
#!/usr/bin/python
# A pattern generator for a holographic volumetric screen
# It takes a series of voxels co-ordinates and colours
# in order to makes sub-holograms based on tile pattern descriptions
# Pygame version for full-screen operation on a dedicated machine
# (c) Holoxica, Oct 2009, written by Javid Khan
#
# release 1.0, Oct 2009
# initial release
#
# release 1.1, Nov 1009
# updated for use with laser projector, refactored code
#
# release 1.2
# added -f fullscreen option
# added configuration options for Width and Height of screen in .cfg file
# note M in .cfg file means size of tiles
#
# release 1.3, Apr 2010
# fixed bug on loading config files
# fixed bug on resizing tiles
#
# release1.4, May 2010
# added pixel scaling factors for non-square pixels (scaleX, scaleY) in cfg
# file
#
# release1.5, Jun 2010
# Pixel scaling factors for non-square pixels (scaleX, scaleY) from keyboard
# y(scaleY + 0.005), u(scaleY - 0.005) & x(scaleX + 0.005), c(scaleX – 0.005)

import pygame
from pygame.locals import *
from pygame.color import THECOLORS
import sys
from patterns import *
import config
from math import *

class testPatterns(object):
    "Pygame version of the pattern generator"

    def __init__(self):
        WINSIZE = (848,480)
        pygame.init()
        self.seqfile, self.cfgfile = 'count2x2.seq', 'default.cfg'
        self.p = Patterns(tkFlag=False)
        print "keyboard commands:\n \
Arrow keys: offset (x and y) \n \
< > keys: margin spacings  \n \
- + keys: size of tile\n g: grid on/off \n \
k: keystone on/off, a & z: keystone up/down \n \
d: dump configuration parameters to a file \n \
s: sequence ON/OFF (timed) \n \
ESC/q: quit \n y&u: scaleY +/- 0.005 \n \
x&c: scaleX +/- 0.005 \n"

        sys.argv.append(self.cfgfile)
        if len(sys.argv)>1:
            params = self.loadParams()
            if params: WINSIZE = (self.p.W, self.p.H)
        if '-f' in sys.argv: self.screen =
pygame.display.set_mode(WINSIZE,pygame.FULLSCREEN)    # fullscreen mode option
        else: self.screen = pygame.display.set_mode(WINSIZE,0,8)  # Windowed
mode
        self.screen.fill(THECOLORS["black"])
        self.p.changeTiles(self.seqfile)
        self.grid, self.keyflag = False, False
        self.p.createGridPoints()
        self.displayMaskPattern()
```

```
        self.main()

    def loadParams(self):
        "load sequences and configuration parameters from .seq and .cfg
files"
        sqfile, cgfile = [i for i in sys.argv if '.seq' in i], [j for j in
sys.argv if '.cfg' in j]
        if sqfile:
            if sqfile[0] in config.getFiles(ext='seq'): self.seqfile =
sqfile[0]
            else:
                print 'error in loading:'+sqfile[0]
                exit()
        defaultParams = {}
        if cgfile:
            if cgfile[0] not in config.getFiles(ext='cfg'):
                print 'error in loading:'+cgfile[0]
                exit()
            else:
                sections = readConfigFile(cgfile[0])
                self.cfgfile = cgfile[0]
                if sections:
                    defaultParams =
parseSection(sections,'default',dictListFlag=False)
                    if defaultParams:
                        for i in defaultParams.keys():    # basic error
checking to ensure params exist
                            if i in Patterns.__dict__.keys():
self.p.__dict__[i]=defaultParams[i]  # assign parameters to class
        return defaultParams

    def dumpConfig(self):
        '''Dump configuration parameters to a file'''
        params = {'W':self.p.W, 'H':self.p.H, 'offsetX':self.p.offsetX,
'offsetY':self.p.offsetY, 'margin':self.p.margin, 'M':self.p.M,
'scaleX':self.p.scaleX, 'scaleY':self.p.scaleY}
        cfgnames = [i.split('.')[0] for i in config.getFiles(ext='cfg') if
'-' in i]
        if not cfgnames:
            cfgnames.append(self.cfgfile.split('.')[0]+'-0')
        cfgnames.sort()
        d = int(cfgnames[-1].split('-')[-1])+1   #increment default file
        fname = 'default-'+str(d)+'.cfg'
        print 'dumping configuration parameters to file:'+fname
        config.writeConfigFile(fname,params)

    def setParams(self,offsetX=0, offsetY=0, margin=0, next=0, m=0, scaleX=0,
scaleY=0):
        "set the main parameters"
        if offsetX+offsetY:
            self.p.offsetX += offsetX
            self.p.offsetY += offsetY
     if scaleX+scaleY:
            self.p.scaleY += scaleY
         self.p.scaleX += scaleX
         self.p.createGridPoints()
               self.displayMaskPattern()
     else:
            self.p.margin += margin
            if self.p.M in range (1,15): self.p.M += m
            else: self.p.M = 11
            self.p.createGridPoints()
        if next: self.p.nextMaskPattern()
        if self.grid: self.displayGrid()
        elif self.keyflag: self.keystone()
        else: self.displayMaskPattern()

    def displayMaskPattern(self):
        "show the mask patterns on the screen"
        X, Y = self.p.offsetX, self.p.offsetY
```

```
            pattern = self.p.currentSequence[self.p.currentSequenceIndex][0]
            colour =
THECOLORS[self.p.currentSequence[self.p.currentSequenceIndex][1]]
            if pattern == 'cls':
                self.screen.fill(colour)
            else:
                self.screen.fill(THECOLORS["black"])  # clear the screen
                for m in self.p.masks[pattern]:
                    square = (m[0]+X,m[1]+Y,self.p.M,self.p.M)
                    pygame.draw.rect(self.screen,colour,square)
            pygame.display.update()

        def displayGrid(self):
            "display a grid on the screen for alignment"
            W,H, X, Y, margin, M = self.p.W, self.p.H, self.p.offsetX,
self.p.offsetY, self.p.marginX, self.p.M
            self.screen.fill(THECOLORS["black"])  # clear the screen
            colour = THECOLORS['green']
            for x in range(0,W,M+margin+X):
                startpos, endpos = (x,0), (x,H)
                pygame.draw.line(self.screen,colour,startpos,endpos)
            for y in range(0,H,M+margin+Y):
                startpos, endpos = (0,y), (W,y)
                pygame.draw.line(self.screen,colour,startpos,endpos)
            pygame.display.update()

        def keystone(self, k=0):
            "display a keystoned mask pattern"
            if not self.keyflag: return
            alpha = pi/20.0
            pattern = self.p.currentSequence[self.p.currentSequenceIndex][0]
            W,H, X, Y, margin, M = self.p.W, self.p.H, self.p.offsetX,
self.p.offsetY, self.p.marginX, self.p.M
            self.screen.fill(THECOLORS["black"])  # clear the screen
            colour = THECOLORS['green']
            for m in self.p.masks[pattern]:
                x, y = m[0], m[1]
                dx = (H-y)*tan(alpha)
                if x>W/2: dx=-dx
                square = (x+dx,y,M,M)
                pygame.draw.rect(self.screen,colour,square)
            pygame.display.update()

        def toggleGrid(self):
            "turn the gridlines on or off"
            self.grid = not self.grid
            if self.grid: self.displayGrid()
            else: self.displayMaskPattern()

        def toggleKeystone(self):
            "turn the keystone correction on or off"
            self.keyflag = not self.keyflag
            if self.keyflag: self.keystone()
            else: self.displayMaskPattern()

        def runSequence(self):
            "timed sequence - change every second"
            if self.p.seqFlag:
                self.p.nextMaskPattern()
                self.displayMaskPattern()

        def toggleSequence(self):
            "toggle the timing sequencer"
            self.p.seqFlag = not(self.p.seqFlag)
            if self.p.seqFlag:
                pygame.time.set_timer(USEREVENT+1, 1000)  # activate timer for
1s = 1000 msec
            else:
                pygame.time.set_timer(USEREVENT+1, 0)  # cancel timer
```

```
    def main(self):
        "mainloop handler for events"
        keyFuncs = {     K_UP: lambda z=-1: self.setParams(offsetY=z),
                         K_DOWN: lambda z=1: self.setParams(offsetY=z),
                         K_LEFT: lambda z=-1: self.setParams(offsetX=z),
                         K_RIGHT: lambda z=1: self.setParams(offsetX=z),
                         K_COMMA: lambda z=-1: self.setParams(margin=z),
                         K_PERIOD: lambda z=1: self.setParams(margin=z),
                         K_SPACE: lambda z=1: self.setParams(next=z),
                         K_EQUALS: lambda z=1: self.setParams(m=z),
                         K_MINUS: lambda z=-1: self.setParams(m=z),
                         K_g: self.toggleGrid,
                         K_k: self.toggleKeystone,
                         K_a: lambda z= 1: self.keystone(k=z),
                         K_a: lambda z=-1: self.keystone(k=z),
                         K_d: self.dumpConfig,
                         K_s: self.toggleSequence,
                         K_y: lambda z = 0.005: self.setParams(scaleY=z),
                         K_u: lambda z = -0.005: self.setParams(scaleY=z),
                         K_x: lambda z = 0.005: self.setParams(scaleX=z),
                         K_c: lambda z = -0.005: self.setParams(scaleX=z)
                   }
        done = False
        while not done:
            # Event Handling:
            events = pygame.event.get( )
            for e in events:
                if e.type == QUIT:
                    done = True
                    break
                elif e.type == USEREVENT+1:
                    self.runSequence()
                elif e.type == KEYDOWN:
                    #pygame.draw.rect(screen,colour,randSquare)
                    k = e.key
                    #print e,'\n',k,'\n'
                    if k in [K_ESCAPE, K_q ]:
                        done = True
                        break
                    elif k in keyFuncs.keys(): keyFuncs[k]()

if __name__=="__main__":
    tp = testPatterns()
```

Appendix C – Scalar Diffraction Theory

This appendix presents an overview of scalar diffraction theory, which describes the propagation of a plane wave between a pair of parallel planes in spaces separated by distance *z* with planar axes *(x,y)*. The theory forms the basis of Fourier Optics which is a valuable tool in modelling and characterising the propagation or diffraction of light in an optical system under certain conditions. Specifically, the theory is concerned with mapping the propagation of a point source in the source plane $p_0(x_0,y_0;0)$ and the observation plane $p_1(x,y;z)$.

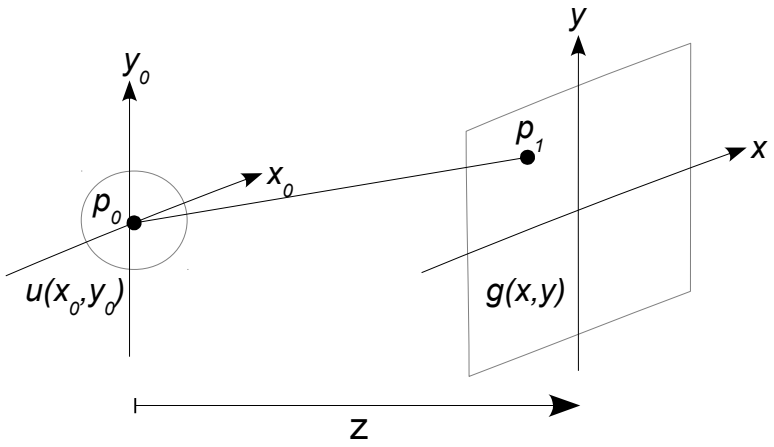

Figure 91: Propagation model

The theory says that a vector light field can be represented simply by a complex (amplitude and phase) scalar potential, derived from Maxwell's equations that describe the interaction between the Electric field, *E*, and the magnetic field, *B*, for an electromagnetic wave:

$$\nabla \cdot E = 0 \qquad \nabla \cdot B = 0 \qquad \nabla \times E = -\frac{\partial B}{\partial t} \qquad \nabla \times B = \varepsilon_0 \mu_o \frac{\partial E}{\partial t} \tag{89}$$

For a monochromatic wave travelling in a linear, isotropic and homogenous medium such as free space, Maxwell's relations lead to the well-known wave equation:

$$\nabla^2 E - \frac{1}{c^2}\frac{\partial^2 E}{\partial t^2} = 0 \tag{90}$$

The starting point of scalar diffraction theory assumes that the E-field can be represented by a complex scalar potential $\psi(r,t)$ which is function of the spatial co-ordinates, *r*, and time, *t*, so the wave equation can be written as:

$$\left(\nabla^2-\frac{1}{c^2}\frac{\partial^2}{\partial t^2}\right)\psi(r,t)=0 \tag{91}$$

The scalar potential can be separated into spatial and time-varying components $\psi(r,t)=u(r)\Phi(t)$ by substitution into the wave equation:

$$\frac{\nabla^2 u(r)}{u(r)}=\frac{1}{c^2\Phi}\frac{\partial\Phi^2}{\partial t^2} \tag{92}$$

It can be seen that the left-hand side represents the spatial field and the right-hand side is the temporal component. Hence, the spatial and temporal components can be considered to be equivalent by setting each side to a constant separator, $-k^2$, thereby enabling the two sides to be handled almost independently as separate functions of t or r, thus:

$$\begin{gathered}\frac{\nabla^2 u(r)}{u(r)}=\frac{1}{c^2\Phi}\frac{\partial\Phi^2}{\partial t^2}=-k^2 \\ \text{i.e. } \frac{\nabla^2 u(r)}{u(r)}=-k^2 \quad \text{or} \quad \frac{1}{c^2\Phi}\frac{\partial\Phi^2}{\partial t^2}=-k^2\end{gathered} \tag{93}$$

The spatial function on the LHS is known as the Helmholtz equation where k is the wave vector. The complex scalar potential can be written in complex amplitude and phase form $U(r)=A(r)\exp(jkr)$, so:

$$\begin{gathered}\nabla^2 u(r)+k^2 u(r)=0 \quad \text{or} \quad \left[\nabla^2+k^2\right]U(r)=0 \\ \left[\nabla^2+k^2\right]A(r)\exp(jkr)=0 \quad \text{where} \quad |k|=\frac{2\pi}{\lambda}\end{gathered} \tag{94}$$

The Helmholtz equation may be solved for $u(r)$ using Green's Theorem, Hecht [145, chap. Appendix 2] through equation (95):

$$g(r)=\frac{1}{4\pi}\left[\oiint_S \frac{\exp(jkr)}{r}\nabla u\,dS-\oiint_S u\nabla\left[\frac{\exp(jkr)}{r}\right]dS\right] \tag{95}$$

This is also known as the Kirchoff Integral Theorem [95, p. 418], where S is the surface bounding the observation plane. The solution leads to the Rayleigh-Sommerfield diffraction integral:

$$g(x,y;z)=\iint_{p_0} u_0(x_0,y_0;0)\frac{\partial}{\partial z}\left(\frac{\exp(jkr)}{r}\right)dx\,dy \qquad \text{where} \quad r=\sqrt{x^2+y^2+z^2} \tag{96}$$

Expansion of the derivative yields:

$$g(x,y;z)=-\frac{1}{\lambda}\iint_{p_0} u_0(x_0,y_0;0)\left[\frac{1}{kr}-j\right]\frac{z}{r}\frac{\exp(jkr)}{r}dx\,dy \tag{97}$$

This describes the scalar potential in the observation plane, *g(x,y;z)* due to the propagation of a single point source p_0 at $u(x_0,y_0;0)$ from the source plane, Figure 91, separated by a distance *r*. Note the field falls as a function of r^2, thus obeying the inverse-square law.

Kirchhoff Diffraction Integral

In general, we are looking at some distance away from the source plane and $z>>\lambda$, so if $r>20\lambda$ then the term $\left[\frac{1}{kr}-j\right]\approx -j$ is simplified, leading to the Kirchoff diffraction integral:

$$g(x,y;z)=-\frac{1}{j\lambda}\iint_{p_0} u_0(x_0,y_0;0)\frac{z}{r}\frac{\exp(jkr)}{r}dx_0\,dy_0 \tag{98}$$

Equation (98) can also be written as a convolution between the source image u_0 and *h(x,y;z)* where * is the convolution operator:

$$g(x,y;z)=-\frac{1}{j\lambda}\iint u_0(x_0,y_0;0)h(x-x_0,y-y_0;z)dx_0\,dy_0$$
$$g(x,y;z)=u_0(x,y;0)*h(x,y;z) \quad \text{where} \quad h(x,y;z)=\frac{1}{j\lambda}\frac{z}{r}\frac{\exp(jkr)}{r} \tag{99}$$

Note that, in general, the distance $r=\sqrt{(x-x_0)^2+(y-y_0)^2+z^2}$. The term *h(x,y;z)* is known as the impulse response of free space, with a constant phase factor $\frac{1}{j\lambda}$, an

obliquity factor $\frac{z}{r}$ and a spherical wave emitter $\frac{\exp(jkr)}{r}$. The constant phase factor can be ignored in calculations and the obliquity factor is sometimes written as an angle $\cos\theta=\frac{z}{r}$ e.g. by Goodman [58]. The Kirchhoff diffraction integral is usually the starting point of Fourier Optics that uses two dimensional signal processing based on numerical Fourier Transform methods to compute digital holograms.

Kirchhoff Angular spectrum

The convolution operation from equation (99) is in the spatial domain and can be reduced to a simple multiplication in the frequency domain.

$$G(f_x, f_y; z)=U(f_x, f_y; 0)\,H(f_x, f_y; z) \tag{100}$$

where f_x and f_y are spatial frequencies $H(f_x f_y)$ is known as the angular spectrum of free space. The Kirchoff angular spectrum is derived by expanding the Helmholtz equation (94) and taking the Fourier Transform:

$$\mathcal{F}\left\{\frac{\partial^2 u}{\partial x^2}+\frac{\partial^2 u}{\partial y^2}+\frac{\partial^2 u}{\partial z^2}+k_0^2 u=0\right\} \;\Rightarrow\; \frac{\partial^2 U}{dz^2}+k^2\left(1-\frac{k_x^2}{k_0^2}-\frac{k_y^2}{k_0^2}\right)=0 \tag{101}$$

Where k_x and k_y are angular spatial frequencies and k_0 is the original k-vector in the same direction of propagation. Equation (101) can be solved to give:

$$G(f_x, f_y; z) \;=\; U(f_x, f_y; 0)\exp\left(-jkz\sqrt{1-\lambda^2(f_x^2+f_y^2)}\right) \tag{102}$$

With f_x and f_y denoting linear spatial frequencies where $k_x=2\pi f_x$ and $k_y=2\pi f_y$ with the field at z given by G and the initial field U. The spatial frequency response or angular spectrum is defined as the ratio of output to input field:

$$H(f_x, f_y; z) \;=\; \frac{G(f_x, f_y; z)}{U(f_x, f_y; 0)} \;=\; \exp\left(-jkz\sqrt{1-\lambda^2(f_x^2+f_y^2)}\right) \tag{103}$$

The linear spatial frequencies are expressed in units of m^{-1} or, more commonly, scaled to lines per mm.

Fresnel Diffraction Integral

A series of geometrical approximations can be applied to the Kirchhoff Diffraction

Integral to make it more palatable. If we assume that the distance between the two planes is much greater than the extent or size of the points or aperture to be imaged i.e. $z^2 \gg \left|x_0^{max} - x_0^{min}\right|^2 + \left|y_0^{max} - y_0^{min}\right|^2$ where the two planes are separated such that $r \approx z$ which is the paraxial approximation applied to equation (98):

$$g(x, y; z) \approx -\frac{1}{j\lambda z}\iint_{P_0} u_0(x_{0,} y_0; 0)\exp(jkr)\,dx_0\,dy_0 \tag{104}$$

and the binomial series expansion is applied to the square root function, *r(x,y,z)*:

$$r = z\sqrt{1+\frac{(x-x_0)^2+(y-y_0)^2}{z^2}} \qquad \sqrt{1+\alpha} = 1+\frac{\alpha}{2!}+\frac{\alpha^2}{3!}+\ldots \tag{105}$$

Taking the first-order term gives:

$$r \approx z\left[1+\frac{(x-x_0)^2+(y-y_0)^2}{2z^2}\right] = z+\frac{(x^2+y^2)+(x_0^2+y_0^2)-2(x\,x_0+y\,y_0)}{2z} \tag{106}$$

Using the series approximation effectively treats the spherical wavefront as a quadratic wavefront. Substituting these approximations in equation (104) gives the Fresnel Diffraction Integral:

$$\begin{aligned} g(x, y; z) &= \frac{1}{j\lambda z}\exp(jkz)\exp\left(\frac{jk}{2z}(x^2+y^2)\right)\times \\ \iint &\left\{u_0(x_{0,} y_0)\exp\left(\frac{jk}{2z}(x_0^2+y_0^2)\right)\right\}\exp\left(\frac{-jk}{z}(x\,x_0+y\,y_0)\right)dx_0\,dy_0 \end{aligned} \tag{107}$$

The field is expressed as the product of a quadratic chirp function and the Fourier transform of the source image multiplied by the same quadratic chirp. The Fourier spatial frequency variables can be written as $f_x = \frac{x}{\lambda z}$ and $f_y = \frac{y}{\lambda z}$ so the Fourier transform integral in equation (107) is:

$$\iint \left\{u_0(x_{0,} y_0)\exp\left(\frac{jk}{2z}(x_0^2+y_0^2)\right)\right\}\exp\left(-j2\pi(f_x x_0+f_y y_0)\right)dx_0\,dy_0 \tag{108}$$

The corresponding Fresnel impulse response of free space is expressed as:

$$h(x, y; z) = \frac{1}{j\lambda z}\exp(-jkz)\exp\left(\frac{-jk(x^2+y^2)}{2z}\right) \quad (109)$$

Fresnel angular spectrum

The general expression in equation (100) applies to all types of propagation including Fresnel. It is derived by taking the 2D Fourier transform of equation (109) or by similarly applying a binomial expansion to the square root in the frequency domain Kirchhoff expression (103) by asserting $\lambda^2 \ll f_x^2 + f_y^2$ i.e. the extent of the total spectral bandwidth is much smaller than the wavelength. The Fresnel angular spectrum is:

$$H(f_x, f_y : z) = \exp(jkz)\exp\left(j\lambda\pi z(f_x^2+f_y^2)\right) \quad (110)$$

Appendix D – Matlab Function Library

This appendix presents a subset of the various Matlab functions that underline FOURIOT, the Fourier Optics Tool, libraries and other programs including propagation models and various holographic displays. These algorithms were initially inspired by a number of Matlab books around Fourier-Optical modelling, notably Voelz, Schmidt, Kim and Poon [96,109,110,146].

Presentation of the routines is summarised in the tables below, and the function header with parameters is shown underneath, including a brief description of the parameters. Most of these can also be found under the corresponding help function within Matlab.

The library is grouped according to the following:

- Source shape functions to control the input shape e.g. solid shapes, lines etc
- Image manipulation: segments, loading images, text functions
- Core functions: 2D FFT/IFFT, matrix, phase functions, SNR, I/O
- Propagation models: Kirchofff, Fraunhoffer, impulse response, angular spectrum

There are many other routines within the codebase but these functions are a representative sample.

Source shape functions

Name	Description
char2imseq	Convert characters to a bitmap
circle	Draw a filled circle
gaussian	Gaussian amplitude distribution
impulseGrid	Make impulses for voxel expansion
linesPlus	Draw a '+' symbol through the origin
linesX	Draw an 'X' shape through the origin
nineseg	Popular seven-segment style digit with + and - symbols
perimeter	Draw an outline of a square
rect2	Draw a filled rectangle
ring	Draw an outline of a circle
sevenseg	Popular 7-segment style digit

```
Function [ out ] = char2imseq(txt,imsize)
%CHAR2IMSEQ convert individual characters in a string text to bitmap images
%   Parameters:
%       txt:    text string
%       imsize: dimensions of image in pixels (square image)
%       out:    normalised of type single float

function [ out ] = circle( X, Y, r )
% circle shape function
%   Parameters:
%       X,Y: meshgrid co-ords
%     scale: scale factor (optional)
%       out: perimeter

function [ out ] = gaussian( X, Y, r, limit )
% 2D gaussian distribution function
%   Parameters:
%       X, Y: Meshgrid co-ords
%          r: size
%      limit: zero threshold, values above this=0
%        out: gaussian array


function [ out ] = impulseGrid( X, grid, gridflag, in, Thresh )
%impulseGrid Make a grid of impulses for voxel expansion
% impulses are places equidistantly across the source plane
% Parameters:
%       X:          Meshgrid co-ords for src plane
%       grid:       [P Q] grid of impulses as [rows cols]
%       gridflag:   flag to indicate full grid or perimeter
%       in:         input signal (optional)
%       Thresh:     threshold value to clear out low signal levels
%       out:        impulses or expanded input signal

function [ out ] = linesPlus( X, Y, r )
%LinesPlus Draw + shaped lines through the origin of size r
%   X,Y: Meshgrid co-ords
%     r: size in m

function [ out ] = linesX( X, Y, r )
%LinesX Draw X-shaped lines through the origin of size r
% Parameters:
```

```
%   X,Y: Meshgrid co-ords
%     r: size in m

function [ out ] = nineseg( X,Y,w, offsetXY, phaseflag, squash )
%NINESEG generate a 2 seg '+' with 7-segment style image
%   Parameters:
%       X,Y : meshgrid co-ords
%       w: size in m
%       offsetXY: offset from centre [offsetX offsetY]
%       phaseflag: random phase ON/OFF
%       squash: 1=squash all images together into one image

function [ out ] = perimeter( X, Y, scale )
% square perimeter function
%   Parameters:
%       X,Y: meshgrid co-ords
%     scale: scale factor (optional)
%       out: perimeter

function [ out ] = rect2( X, Y, scale )
%RECT2 improved square generator function
%   Parameters:
%       X,Y: meshgrid co-ords
%     scale: scale factor (optional)
%       out: square

function [ out ] = ring( X, Y, r )
% generate a ring
%   Parameters:
%       X,Y: meshgrid co-ords
%         r: ring radius
%       out: perimeter

function out = sevenseg( X,Y,w, offsetXY, phaseflag, squash )
%SEVENSEG generate a 7-segment style image
%   Parameters:
%       X,Y : meshgrid co-ords
%       w: size in m
%       offsetXY: offset from centre [offsetX offsetY]
%       phaseflag: random phase ON/OFF
%       squash: 1=squash all images together into one
```

Image manipulation

Name	Description
generateSegment	Generate a single horizontal or vertical segment for a seven-segment style digit.
getImage	Load an image from disc and insert into a matrix
im2points	Convert a 2D image matrix into a series of 1D points
intshow	Show the intensity of an image
phaseim	Plot the phase of a complex input
text2mat	Convert a text string into a 2D matrix
text2matseq	Convert a text string into a 3D matrix of characters
textMov	Covert a text string into a movie sequence of characters

```
function out = generateSegment( X,Y,w,offset,type)
%SEVENSEG generate a 7-segment style image
%   Parameters:
%       X,Y :   meshgrid co-ords
%       w:      size in m
%       type:   'vertical' or 'horizontal' or 'v' or 'h'
%       offsetXY: [offsetX offsetY]
%       out:    output segments

function [ out ] = getimage( in, fname )
%GETIMAGE retrieve an image, scale and insert into a matrix
%   Parameters:
%       in:  input matrix
%      out:  image inserted into input matrix

function [ out ] = im2points( X, Y, in, z, thresh )
%IM2POINTS Convert an image to a set of points
%   Points characterised in terms of ampitude, position and phase (if
%   applicable). This is useful for pointwise convolution in CGH algorithms
%   Parameters:
%      X,Y: Meshgrid co-ords
%       in: 2D image array, can be complex
%        z: 2D image depth
%   thresh: threshold, default=.001
%      out: [a x y z p], where a:amplitude, [x y z]:co-ords & p:phase 0-2PI

function  intshow( in )
%INTSHOW show the intensity of a 2D function
%   Parameters
%       in: input (2D) array, can be complex

function phaseim ( in )
%PHASEIM Plot phase of complex input
%   Shifts the phase from -PI..PI to 0..2PI and converts to greyscale
%   Parameters:
%           in: complex number
%         flag: show image
%          out: grayscale image

function [ out ] = text2mat( txt, dim, offset, scale, phaseflag, dsm )
%TEXT2MAT make a series of images of the input text as a matrix sequence
%   out is normalised of type single float
%   Parameters:
%       txt: text string
%       dim: dimensions of image [rows cols]
%       offest: image offsets as %age [startX startY] e.g. [0.2 0.9]
%       scale: scale factor, to avoid hitting the edge, default=0.9
%       phaseflag: add random phase
%       dsm: downsampling matrix [P Q]

function [ out ] = text2matseq( txt, dim, offset, scale, phaseflag, dsm )
%TEXT2MAT make a sequence of images of the input text as a matrix sequence
%   out is normalised of type single float
%   Parameters:
%      txt: text string
%      dim: dimensions of image [rows cols]
%      offest: image offsets as %age [startX startY] or [startX startY dx dy]
%                e.g. [0.2 0.9 0 0], if only 2 params then auto calc dx dy
%      scale: scale factor, to avoid hitting the edge, default=0.9
%      phaseflag: add random phase
%      dsm: downsampling matrix [P Q] e.g. [2 2]=2x2 tile

function [ out ] = textMov( txt, dim )
%TEXTMOV make a movie out of the input text
%   Parameters
%       txt: text string
%       dim: size of output array
%       out: bitmap containing text grabbed from screen
```

Core support functions

Name	Description
ft2	2D Fast Fourier Transform with optical alignment
ift2	Inverse 2D FFT
inflate	Resize a matrix
insertMat	Insert a smaller matrix into a larger one (centred)
randphase	Random phase matrix with optional quantisation
rms	Root mean square
shiftphase	Shift phase from -PI..+PI to 0..2PI
showAll	Take a 3D image array and squashes it to a 2D image
shuffle	Randomly rearrange the elements of an array
snr	Signal-to-noise ratio
writephase	Write phase to a file as a greyscale bitmap

```
function [ out ] = ft2( in )
%2D Fourier Transform with shifting for correct optical alignment
% using the fftshift function
%   Parameters:
%       in: input source, must be 2D
%      out: output, aligned complex fft

function [ out ] = ift2( in )
% inverse 2D Fourier Transform with shifting for correct optical alignment
% using the fftshift function
%   Parameters:
%       in: input source, must be 2D
%      out: output, aligned complex inverse fft

function [ out ] = inflate(in, reSize, phaseflag)
%INFLATE resize input matrix
%   Parameters:
%           in: input matrix
%       reSize: resizing factor e.g. 1.5
%          out: inflated input matrix

function [ out ] = insertMat(in,ins)
%INSERTMAT insert a smaller matrix into a larger one
% smaller matrix is centred within larger one, this is useful for
% zero-padding images
%   Parameters:
%       in: input matrix
%      ins: insert matrix (smaller than in)
%      out: centred matrix

function [ out ] = randphase( in, levels )
%RANDPHASE Add random phase to a signal with optional phase quantization
%   Parameters:
%         in: input signal
%     levels: quantization levels
%        out: quantized input signal

function out = rms( input )
%RMS compute the root mean square of the input
%   Parameters
%       in: input array, can be complex
```

```
%      out: rms value, not normalised

function [ out ] = shiftphase( phase )
%SHIFTPHASE phase shifting a complex matrix
%   By adding 2PI to all negative phase values
%   Parameters
%       in: input complex phase angle values -PI..+PI
%      out: output phase values 0..2PI

function [ out ] = showAll( inseq, flag )
%showAll Show all image sequences in image sequence array
%   Parameters:
%       inseq: sequence of MxN x nr images
%       flag: show the combined image
%       out: all images squashed together

function [ out ] = shuffle( in )
% SHUFFLE shuffle the input vector or matrix in a random way
%   Parameters
%       in: input array
%      out: shuffled array

function signoise = snr( signal, measured )
%SNR Compute signal-to-noise ratio in dB
%   input signal and measured signals as parameters (can be complex values)
%   Parameters
%       signal: input or ideal signal, can be complex
%       measured: measured signal with noise
%       out:    SNR as a scalar in dB

function writephase( in, fname )
%writephase write the phase of a complex number into a bitmap
%   Parameters:
%       in: input matrix
%       fname: filename
```

Propagation models

Name	Description
ConvProp2	Direct convolution (Kirchoff model)
forwardProp	Forward propagation model
HrsIR	Kirchoff impulse response propagator
HrsAS	Kirchoff angular spectrum propagator
HfrIR	Fresnel impulse response propagator
HfrAS	Fresnel angular spectrum propagator
reverseProp	Reverse propagation model
SingleParallax	Direct convolution vertical or horizontal parallax only

```
function [ out ] = convProp2( pts, M, N, k, dx )
%CONVPROP2 Krichoff Propagation via convolution of a set of points in space
%   with the hologram plane. Faster optimised version based on the impulse
%   displacement method where a larger hologram plane is pre-computed and
%   then displaced & modified with respect to each point
%
%   Parameters:
%       pts: point cloud co-ords [a x y z p], where a:ampl, p:phase
%       M,N: size of hologram plane
%         k: k-vector = 2*pi/lambda
```

```
%        dx: holopixel size
%       out: complex convolution result

function [ out ] = forwardProp( g, H )
%forwardProp forward propagation of field g with propagation function H
% H can be based on any of Rayleigh Sommerfied, Fresnel or Fraunhoffer
% propagation methods
% Parameters:
%       g : spatial domain input
%       H : transfer function (frequency domain)
%     out : spatial domain output

function [ H ] = HrsAS(FX, FY, z, k)
%HrsAS Rayleigh Sommerfield Kirchoff angular spectrum transfer function
%   Parameters
%           X, Y: meshgrid co-ords
%           z:    propagation distance
%           k:    wave vector
%      return:    Kirchoff IR function

function [ H ] = HrsIR( X, Y, z, k, dx )
%HrsIR Rayleigh Sommerfield impulse response
%   Parameters
%           X, Y: meshgrid co-ords
%           z:    propagation distance
%           k:    wave vector
%      return:    R-S IR function

function [ H ] = HfrAS(FX, FY, z, k)
%HfrAS Fresnel angular spectrum transfer function
%   Parameters
%           X, Y: meshgrid co-ords
%           z:    propagation distance
%           k:    wave vector
%      return:    Fresnel AS transfer function

function [ out ] = reverseProp( g, H )
%reverseProp reverse propagation of field g with propagation function H
% H can be based on any of Rayleigh Sommerfied, Fresnel or Fraunhoffer
% propagation methods. H is inverted in this model, useful for impulse
% response functions
%   Parameters:
%       g : spatial domain input
%       H : transfer function (frequency domain)
%     out : spatial domain output

function [ out ] = singleParallax( in, z, k, dx, type)
%singleParallax Generation of a hologram using single-parallax techniques
%   Including horizontal (default) and vertical parallax or both
%   to save on computation time
%   Parameters:
%       in: input source image (can be complex)
%        z: distance between source and hologram plane
%        k: wave vector = 2*pi/lambda
%       dx: holopixel size
%     type: 'horizontal', 'vertical' or 'both'
%      out: hologram plane image
```

References

[1] R. Zone, *Stereoscopic Cinema and the Origins of 3-D Film, 1838-1952*, The University Press of Kentucky (2007).

[2] R. Zone, *3-D Revolution: The History of Modern Stereoscopic Cinema*, University Press of Kentucky (2012).

[3] L. Lipton, *Brief history of electronic stereoscopic displays*, Opt. Eng, **51**(2), 021103–1 (2012) [doi:10.1117/1.OE.51.2.021103].

[4] H. Kolb, R. Nelson, E. Fernandez, and B. W. Jones, *Webvision - The Organization of the Retina and Visual System*, <http://webvision.med.utah.edu/book/> (13 September 2012).

[5] D. M. Hoffman, A. R. Girshick, K. Akeley, and M. S. Banks, *Vergence–accommodation conflicts hinder visual performance and cause visual fatigue*, J Vis, **8**(3), 33.1–3330 (2008) [doi:10.1167/8.3.33].

[6] M. Lambooij, M. Fortuin, I. Heynderickx, and W. IJsselsteijn, *Visual Discomfort and Visual Fatigue of Stereoscopic Displays: A Review*, Journal of Imaging Science and Technology, **53**(3), 30201–1–30201–30214 (2009) [doi:10.2352/J.ImagingSci.Technol.2009.53.3.030201].

[7] T. Q. Khanh and P. Bodrogi, *Illumination, Color and Imaging: Evaluation and Optimization of Visual Displays*, Wiley VCH (2012).

[8] R. S. Berns, *Principles of Colour Technology*, 3rd Edition, Wiley-Blackwell (2000).

[9] F. Yaras, H. Kang, and L. Onural, *State of the Art in Holographic Displays: A Survey*, J. Display Technol., **6**(10), 443–454 (2010).

[10] J. Hong, Y. Kim, H.-J. Choi, J. Hahn, J.-H. Park, H. Kim, S.-W. Min, N. Chen, and B. Lee, *Three-dimensional display technologies of recent interest: principles, status, and issues [Invited]*, Appl. Opt., **50**(34), H87–H115 (2011) [doi:10.1364/AO.50.000H87].

[11] Wikipedia contributors, *Parallax barrier*, Wikipedia, the free encyclopedia, Wikimedia Foundation, Inc. (2012).

[12] D. Brown, *The three dimensions of John Logie Baird*, Radio Society of Great Britain, Bedford (2012).

[13] B. G. Blundell and A. J. Schwarz, *Volumetric Three-dimensional Display Systems*, John Wiley & Sons, Incorporated (2000).

[14] G. E. Favalora, *Progress in Volumetric Three-Dimensional Displays and Their*

Applications, Frontiers in Optics, p. FTuT2, Optical Society of America (2009).

[15] B. G. Blundell, *3D Displays and Spatial Interaction: Exploring the Science, Art, Evolution and Use of 3D Technologies, Vol. 1: From Perception to Technology*, Walker & Wood Limited (2011).

[16] H. Saito, H. Kimura, S. Shimada, T. Naemura, J. Kayahara, S. Jarusirisawad, V. Nozick, H. Ishikawa, T. Murakami, et al., *Laser-plasma scanning 3D display for putting digital contents in free space*, Proc. SPIE, **6803**, pp. 680309–680318, SPIE (2008) [doi:10.1117/12.768068].

[17] M. Dreyer, G. K. Newman, E. Petrich, H. H. Refai, J. J. J. Sluss, M. P. Tull, and P. Verma, *Light surface display for rendering a three-dimensional image*, US7858913 B2 (2010).

[18] S. F. Brown, *Seeing Triple*, Sci Am, **296**(6), 86–86 (2007) [doi:10.1038/scientificamerican0607-86].

[19] G. E. Favalora, *Volumetric 3D Displays and Application Infrastructure*, Computer, **38**(8), 37–44 (2005) [doi:10.1109/MC.2005.276].

[20] G. E. Favalora, *Multiplanar autostereoscopic imaging system*, US5936767 A (1999).

[21] G. E. Favalora, J. Napoli, D. M. Hall, R. K. Dorval, M. Giovinco, M. J. Richmond, and W. S. Chun, *100-million-voxel volumetric display*, 300–312 (2002) [doi:10.1117/12.480930].

[22] A. Sullivan, *Multi-planar volumetric display system and method of operation*, US6100862 A (2000).

[23] A. Sullivan, *DepthCube solid-state 3D volumetric display*, 279–284 (2004) [doi:10.1117/12.527543].

[24] Lippmann, G., *Epreuves reversibles. Photographies integrales*, Comptes Rendus, **146**, 446–451 (1908).

[25] J.-S. Jang and B. Javidi, *Time-Multiplexed Integral Imaging For 3D Sensing and Display*, Opt. Photon. News, **15**(4), 36–43 (2004) [doi:10.1364/OPN.15.4.000036].

[26] T. Balogh, *Method and apparatus for displaying three-dimensional images*, WO1998034411 A1 (1998).

[27] T. Balogh, *The HoloVizio system*, Proc. SPIE, **6055**, A. J. Woods, N. A. Dodgson, J. O. Merritt, M. T. Bolas, and I. E. McDowall, Eds., p. 60550U–12, SPIE, San Jose, CA, USA (2006).

[28] L. Onural, A. Gotchev, H. M. Ozaktas, and E. Stoykova, *A Survey of Signal Processing Problems and Tools in Holographic Three-Dimensional Television*, Circuits and Systems for Video Technology, IEEE Transactions on, **17**(11), 1631–1646 (2007) [doi:10.1109/TCSVT.2007.909973].

[29] H. I. Bjelkhagen, *Special Section Guest Editorial: Digital Holography and Holographic Displays*, Opt. Eng, **50**(9), 091301–091301 (2011) [doi:10.1117/1.3622483].

[30] M. L. Jepsen, *Holographic video: design and implementation of a display system*, Ph.D., Massachusetts Institute of Technology (1989).

[31] P. St-Hilaire, S. A. Benton, M. E. Lucente, and P. M. Hubel, *Color images with the MIT holographic video display*, Proc. SPIE, **1667**, pp. 73–84, SPIE, San Jose (1992) [doi:10.1117/12.59642].

[32] P. St-Hilaire, *Scalable Optical Architectures for Electronic Holography*, Ph.D., Massachusetts Institute of Technology (1994).

[33] P. St-Hilaire, S. A. Benton, M. E. Lucente, J. D. Sutter, and W. J. Plesniak, *Advances in holographic video*, Proc. SPIE, **1914**, pp. 188–196, San Jose, CA, USA (1993) [doi:10.1117/12.155020].

[34] M. E. Lucente, *Optimization of hologram computation for real-time display*, Proc. SPIE, **1667**, pp. 32–43, SPIE, San Jose, CA, USA (1992) [doi:10.1117/12.59617].

[35] C. Slinger, C. Cameron, and M. Stanley, *Computer-Generated Holography as a Generic Display Technology*, Computer, **38**(8), 46–53 (2005) [doi:10.1109/MC.2005.260].

[36] M. Stanley, M. A. Smith, A. P. Smith, P. J. Watson, S. D. Coomber, C. D. Cameron, C. W. Slinger, and A. Wood, *3D electronic holography display system using a 100-megapixel spatial light modulator*, Proc. SPIE, **5249**, pp. 297–308 (2004) [doi:10.1117/12.516540].

[37] M. Huebschman, B. Munjuluri, and H. Garner, *Dynamic holographic 3-D image projection*, Opt. Express, **11**(5), 437–445 (2003).

[38] X. Xu, X. Liang, Y. Pan, R. Zheng, Z. A. Lum, P. P. Mar Yi Lwin, and S. Solanki, *Development of full-color full-parallax digital 3D holographic display system and its prospects*, Proc. SPIE, **8864**, pp. 864409–864409, SPIE, San Francisco (2013) [doi:10.1117/12.2002388].

[39] A. Schwerdtner, R. Haussler, and N. Leister, *Large holographic displays for*

real-time applications, Proc. SPIE, **6912**, p. 69120T–8, SPIE, San Jose, CA, USA (2008).

[40] S. Reichelt and N. Leister, *Computational hologram synthesis and representation on spatial light modulators for real-time 3D holographic imaging*, J. Phys.: Conf. Ser., **415**(1), 012038 (2013) [doi:10.1088/1742-6596/415/1/012038].

[41] S. Tay, P.-A. Blanche, R. Voorakaranam, A. V. Tunc, W. Lin, S. Rokutanda, T. Gu, D. Flores, P. Wang, et al., *An updatable holographic three-dimensional display*, Nature, **451**(7179), 694–698 (2008) [doi:10.1038/nature06596].

[42] P.-A. Blanche, A. Bablumian, R. Voorakaranam, C. Christenson, W. Lin, T. Gu, D. Flores, P. Wang, W.-Y. Hsieh, et al., *Holographic three-dimensional telepresence using large-area photorefractive polymer*, Nature, **468**(7320), 80–83 (2010) [doi:10.1038/nature09521].

[43] H. Gao, X. Li, Z. He, Y. Su, and T.-C. Poon, *59.4: Real-Time Dynamic Holographic Display Based on a Liquid Crystal Thin Film*, SID Symposium Digest of Technical Papers, **43**(1), 804–807 (2012) [doi:10.1002/j.2168-0159.2012.tb05907.x].

[44] H. Gao, X. Li, Z. He, Y. Su, and T.-C. Poon, *Multiplexed holographic display based on a fast response liquid crystal film*, Digital Holography and Three-Dimensional Imaging, p. DM2C.4, Optical Society of America (2012).

[45] M. A. Klug, T. Burnett, A. Fancello, A. Heath, K. Gardner, S. O'Connell, and C. Newswanger, *A Scalable, Collaborative, Interactive Light-field Display System*, SID Symposium Digest of Technical Papers, **44**(1), 412–415 (2013) [doi:10.1002/j.2168-0159.2012.tb05692.x].

[46] V. M. Bove, *Display Holography's Digital Second Act*, Proceedings of the IEEE, **100**(4), 918 –928 (2012) [doi:10.1109/JPROC.2011.2182071].

[47] G. Dennis, *Improvements In And Relating To Microscopy*, GB685286 (1947).

[48] D. Gabor, *A New Microscopic Principle*, Nature, **161**(4098), 777–778 (1948) [doi:10.1038/161777a0].

[49] D. Gabor, *Microscopy by Reconstructed Wave-Fronts*, Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences, **197**(1051), 454 –487 (1949) [doi:10.1098/rspa.1949.0075].

[50] E. N. Leith and J. Upatnieks, *Reconstructed Wavefronts and Communication Theory*, J. Opt. Soc. Am., **52**(10), 1123–1128 (1962)

[doi:10.1364/JOSA.52.001123].

[51] Y. Denisyuk, *On the reflection of optical properties of an object in a wave field of light scattered by it*, Doklady Akademii Nauk SSSR, **144**, 1275–1278 (1962).

[52] S. A. Benton, *Applications of Holography and Optical Data Processing: Proceedings of an International Conference, Jerusalem, 1976*, Applications of Holography and Optical Data Processing: Proceedings of an International Conference, E. Marom, E. Wiener-Avnear, and A. A. Friesem, Eds., pp. 401–409, Pergamon Press Reprint, Jerusalem (1977).

[53] S. A. Benton and V. M. Bove, *Holographic Imaging*, John Wiley & Sons, Inc., Hoboken, NJ, USA (2008).

[54] P. Hariharan, *Basics of holography*, Cambridge University Press (2002).

[55] G. Saxby, *Practical holography*, CRC Press (2004).

[56] H. Kogelnik, *Coupled wave theory for thick hologram gratings*, The Bell System Technical Journal, Vol. 48, no. 9, November 1969, pp. 2909-2947, **48**, 2909–2947 (1969).

[57] B. C. Kress and P. Meyrueis, *Applied Digital Optics: From Micro-optics to Nanophotonics*, Wiley (2009).

[58] J. W. Goodman, *Introduction to Fourier optics*, 3rd ed., Roberts and Company Publishers (2005).

[59] D. Brotherton-Ratcliffe, *A treatment of the general volume holographic grating as an array of parallel stacked mirrors*, Journal of Modern Optics, **59**(13), 1113–1132 (2012) [doi:10.1080/09500340.2012.695405].

[60] D. Brotherton-Ratcliffe, *A New Type of Coupled Wave Theory Capable of Analytically Describing Diffraction in Polychromatic Gratings and Holograms*, J. Phys.: Conf. Ser., **415**(1), 012034 (2013) [doi:10.1088/1742-6596/415/1/012034].

[61] B.-R. David, *Understanding Diffraction in Volume Gratings and Holograms*, Holography - Basic Principles and Contemporary Applications, E. Mihaylova, Ed., InTech (2013).

[62] D. J. DeBitetto, *Holographic Panoramic Stereograms Synthesized from White Light Recordings*, Appl. Opt., **8**(8), 1740–1741 (1969) [doi:10.1364/AO.8.001740].

[63] B. R. Brown and A. W. Lohmann, *Complex Spatial Filtering with Binary Masks*, Appl. Opt., **5**(6), 967–969 (1966) [doi:10.1364/AO.5.000967].

[64] L. B. Lesem, P. M. Hirsch, and J. A. Jordan, *The kinoform: a new wavefront reconstruction device*, IBM J. Res. Dev., **13**(2), 150–155 (1969).

[65] R. W. Gerchberg and W. O. Saxton, *A practical algorithm for the determination of phase from image and diffraction plane pictures*, Optik, **35**, 227–246 (1972).

[66] G. A. Mills and I. Yamaguchi, *Effects of quantization in phase-shifting digital holography*, Appl. Opt., **44**(7), 1216–1225 (2005) [doi:10.1364/AO.44.001216].

[67] K. S. Urquhart, R. Stein, and S. H. Lee, *Computer-generated holograms fabricated by direct write of positive electron-beam resist*, Opt. Lett., **18**(4), 308–310 (1993) [doi:10.1364/OL.18.000308].

[68] G. K. Ackermann and J. Eichler, *Holography*, Wiley-VCH (2007).

[69] H. I. Bjelkhagen, *Ultra-realistic 3-D imaging based on colour holography*, J. Phys.: Conf. Ser., **415**(1), 012023 (2013) [doi:10.1088/1742-6596/415/1/012023].

[70] J. Harris, G. C. Sherman, and B. H. Billings, *Copying Holograms*, Appl. Opt., **5**(4), 665–666 (1966) [doi:10.1364/AO.5.000665].

[71] W. O. Davis, R. Sprague, and J. Miller, *MEMS-based pico projector display*, Optical MEMs and Nanophotonics, 2008 IEEE/LEOS International Conference on, pp. 31 –32 (2008) [doi:10.1109/OMEMS.2008.4607813].

[72] M. Lutz, *Programming Python*, O'Reilly Media, Inc. (2009).

[73] Beazley, *Python Essential Reference, 3/E*, Pearson Education (2006).

[74] W. McGugan, *Beginning Game Development with Python and Pygame: From Novice to Professional*, Apress (2007).

[75] J. Khan, I. Underwood, A. Greenaway, and M. Halonen, *A low-resolution 3D holographic volumetric display*, Proc. SPIE, **7723**, P. Schelkens, T. Ebrahimi, G. Cristobal, F. Truchetet, and P. Saarikko, Eds., p. 77231B–7, SPIE, Brussels, Belgium (2010) [doi:10.1117/12.858596].

[76] J. Khan, *Three dimensional holographic volumetric display*, US8625183 B2 (2014).

[77] M. C. King, A. M. Noll, and D. H. Berry, *A New Approach to Computer-Generated Holography*, Appl. Opt., **9**(2), 471–475 (1970) [doi:10.1364/AO.9.000471].

[78] M. C. King and M. C. King, *Synthetic Hologram Generation From*, 3832027 (1974).

[79] M. Yamaguchi, T. Koyama, H. Endoh, N. Ohyama, S. Takahashi, and F. Iwata,

Development of a prototype full-parallax holoprinter, Proc. SPIE, **2406**, pp. 50–56, SPIE, San Jose (1995) [doi:10.1117/12.206244].

[80] M. A. Klug, M. W. Halle, M. E. Lucente, and W. J. Plesniak, *Compact prototype one-step Ultragram printer*, Proc. SPIE, **1914**, pp. 15–24, SPIE, San Jose (1993) [doi:10.1117/12.155022].

[81] E. van Nuland, W. C. Spierings, and N. Govers, *Development of an office holoprinter V*, Proc. SPIE, **2652**, pp. 62–69, SPIE, San Jose (1996) [doi:10.1117/12.236046].

[82] F. Yang, Y. Murakami, and M. Yamaguchi, *Digital color management in full-color holographic three-dimensional printer*, Applied Optics, **51**(19), 4343 (2012) [doi:10.1364/AO.51.004343].

[83] W. Plesniak, J., V. Michael Bove, R. Pappu, J. Barabas, and M. Halle, *Reconfigurable image projection holograms*, Opt. Eng, **45**(11), 115801–115801–15 (2006) [doi:10.1117/1.2390678].

[84] D. Brotherton-Ratcliffe and A. Rodin, *Holographic printer*, 7161722 (2007).

[85] H. I. Bjelkhagen and D. Brotherton-Ratcliffe, *Ultra-Realistic Imaging: Advanced Techniques in Analogue and Digital Colour Holography*, Taylor & Francis Group (2013).

[86] *Philosophical Transactions of the Royal Society of London*, The Society, London (1815).

[87] D. Brotherton-Ratcliffe, S. J. Zacharovas, R. J. Bakanas, J. Pileckas, A. Nikolskij, and J. Kuchin, *Digital holographic printing using pulsed RGB lasers*, Optical Engineering, **50**, 091307 (2011) [doi:10.1117/1.3596182].

[88] R. Kingslake, *Applied Optics and Optical Engineering*, Elsevier (2012).

[89] S. Zacharovas, A. Nikolskij, R. Bakanas, and D. Brotherton-Ratcliffe, *One-step 3D full-colour and achromatic transmission holograms digitally printed using a 440nm pulsed laser for embossed applications*, Proc. SPIE, **8644**, pp. 864407–864407, SPIE, San Francisco (2013) [doi:10.1117/12.2008469].

[90] T. Yamaguchi, O. Miyamoto, and H. Yoshikawa, *Volume hologram printer to record the wavefront of three-dimensional objects*, Optical Engineering, **51**(7), 075802 (2012) [doi:10.1117/1.OE.51.7.075802].

[91] G. Tricoles, *Computer generated holograms: an historical review*, Appl. Opt., **26**(20), 4351–4357 (1987) [doi:10.1364/AO.26.004351].

[92] L. Ahrenberg, P. Benzie, M. Magnor, and J. Watson, *Computer generated*

holography using parallel commodity graphics hardware, Opt. Express, **14**(17), 7636–7641 (2006) [doi:10.1364/OE.14.007636].

[93] R. H.-Y. Chen and T. D. Wilkinson, *Computer generated hologram from point cloud using graphics processor*, Appl. Opt., **48**(36), 6841–6850 (2009) [doi:10.1364/AO.48.006841].

[94] G. J. Swanson and L. Laboratory, *Binary optics technology: the theory and design of multi-level diffractive optical elements*, Massachusetts Institute of Technology, Lincoln Laboratory (1989).

[95] M. Born and E. Wolf, *Principles of Optics: Electromagnetic Theory of Propagation, Interference and Diffraction of Light*, Cambridge University Press (1999).

[96] D. G. Voelz, *Computational Fourier Optics: A Matlab Tutorial*, SPIE Press (2011).

[97] H. Nyquist, *Certain Topics in Telegraph Transmission Theory*, American Institute of Electrical Engineers, Transactions of the, **47**(2), 617–644 (1928) [doi:10.1109/T-AIEE.1928.5055024].

[98] C. Shannon, *Communication in the Presence of Noise*, Proceedings of the IRE, **37**(1), 21, 10 (1949).

[99] M. Lucente, *Diffraction-specific fringe computation for electro-holography*, Ph.D., Massachusetts Institute of Technology (1995).

[100] R. Brauer, F. Wyrowski, and O. Bryngdahl, *Diffusers in digital holography*, J. Opt. Soc. Am. A, **8**(3), 572–578 (1991) [doi:10.1364/JOSAA.8.000572].

[101] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 2nd ed., Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA (2001).

[102] P. Birch, R. Young, M. Farsari, C. Chatwin, and D. Budgett, *A comparison of the iterative Fourier transform method and evolutionary algorithms for the design of diffractive optical elements*, Optics and Lasers in Engineering, **33**(6), 439–448 (2000) [doi:10.1016/S0143-8166(00)00044-0].

[103] E. Osherovich, M. Zibulevsky, and I. Yavneh, *Approximate Fourier phase information in the phase retrieval problem: what it gives and how to use it*, Journal of the Optical Society of America A, **28**(10), 2124 (2011) [doi:10.1364/JOSAA.28.002124].

[104] A. Georgiou, T. D. Wilkinson, N. Collings, and W. A. Crossland, *An algorithm for computing spot-generating holograms*, Journal of Optics A: Pure and

Applied Optics, **10**(1), 015306 (2008) [doi:10.1088/1464-4258/10/01/015306].

[105] D. C. O'Shea, *Diffractive Optics: Design, Fabrication, and Test*, SPIE Press (2004).

[106] D. G. Voelz and M. C. Roggemann, *Digital simulation of scalar optical diffraction: revisiting chirp function sampling criteria and consequences*, Appl. Opt., **48**(32), 6132–6142 (2009) [doi:10.1364/AO.48.006132].

[107] P. Lobaz, *Discrete calculation of the off-axis angular spectrum based light propagation*, J. Phys.: Conf. Ser., **415**(1), 012040 (2013) [doi:10.1088/1742-6596/415/1/012040].

[108] F. J. Harris, *On the use of windows for harmonic analysis with the discrete Fourier transform*, Proceedings of the IEEE, **66**(1), 51–83 (1978) [doi:10.1109/PROC.1978.10837].

[109] T.-C. Poon and T. Kim, *Engineering optics with Matlab*, World Scientific (2006).

[110] J. D. Schmidt, *Numerical Simulation of Optical Wave Propagation with Examples in MATLAB*, SPIE Books (2010).

[111] *Pepper's ghost*, Wikipedia, the free encyclopedia (2013).

[112] B. Blundell, *Enhanced Visualization: Making Space for 3-D Images*, John Wiley & Sons (2007).

[113] J. Cech, *Forming of Diffractive Micro-optical Elements in Amorphous Chalcogenide Thin Films*, Ph.D., University of Pardubice (2002).

[114] P. Hariharan, *Optical Holography: Principles, Techniques and Applications*, Second, Cambridge University Press (1996).

[115] M. Gu, *Advanced Optical Imaging Theory*, Springer (1999).

[116] D. Jurbergs, F.-K. Bruder, F. Deuber, T. Fäcke, R. Hagen, D. Hönel, T. Rölle, M.-S. Weiser, and A. Volkov, *New recording materials for the holographic industry*, 2009, 72330K–72330K–10 [doi:10.1117/12.809579].

[117] J. Guild, *The Colorimetric Properties of the Spectrum*, Phil. Trans. R. Soc. Lond. A, **230**(681-693), 149–187 (1932) [doi:10.1098/rsta.1932.0005].

[118] R. W. G. Hunt and M. R. Pointer, *Measuring Colour*, John Wiley & Sons (2011).

[119] A. Martelli, A. Ravenscroft, and D. Ascher, *Python Cookbook*, O'Reilly Media, Incorporated (2005).

[120] D. Padiyar and J. Padiyar, *Aberrations in Holography*, J. Phys.: Conf. Ser.,

415(1), 012031 (2013) [doi:10.1088/1742-6596/415/1/012031].

[121] E. B. Champagne, *Nonparaxial Imaging, Magnification, and Aberration Properties in Holography*, J. Opt. Soc. Am., **57**(1), 51–55 (1967) [doi:10.1364/JOSA.57.000051].

[122] F. T. S. Yu and X. Yang, *Introduction to Optical Engineering*, Cambridge University Press (1997).

[123] S. Yakov, *Field Guide to Diffractive Optics*, SPIE Press (2011).

[124] J. Khan, C. Can, A. Greenaway, and I. Underwood, *A real-space interactive holographic display based on a large-aperture HOE*, Proc. SPIE, **8644**, p. 86440M–86440M, SPIE, San Francisco (2013) [doi:10.1117/12.2021633].

[125] S. A. Benton, *Hologram Reconstructions with Extended Incoherent Sources*, J. Opt. Soc. Am., **59**, 1545A (1969).

[126] S. A. Benton, J. Mingace, and W. R. Walter, *One-Step White-Light Transmission Holography*, 156–161 (1980) [doi:10.1117/12.958435].

[127] K. A. Stetson, *Holography with total internally reflected light*, Applied Physics Letters, **11**(7), 225–226 (1967) [doi:10.1063/1.1755109].

[128] L. H. Lin, *Edge-illuminated hologram*, J. Opt. Soc. Am., **60**, 714A (1970).

[129] S. A. Benton, S. M. Birner, and A. Shirakura, *Edge-lit rainbow holograms*, Proc. SPIE, **1212**, pp. 149–157, SPIE, Los Angeles (1990) [doi:10.1117/12.17977].

[130] D. Leseberg and O. Bryngdahl, *Computer-generated rainbow holograms*, Appl. Opt., **23**(14), 2441–2447 (1984) [doi:10.1364/AO.23.002441].

[131] E. Bressert, *SciPy and NumPy*, O'Reilly Media, Inc. (2012).

[132] M. M. Waldrop, *Brilliant Displays*, Scientific American, **297**(5), 94–97 (2007) [doi:10.1038/scientificamerican1107-94].

[133] M. W. Miles, *Visible spectrum modulator arrays*, US5835255 (A) (1998).

[134] D. Fattal, Z. Peng, T. Tran, S. Vo, M. Fiorentino, J. Brug, and R. G. Beausoleil, *A multi-directional backlight for a wide-angle, glasses-free three-dimensional display*, Nature, **495**(7441), 348–351 (2013) [doi:10.1038/nature11972].

[135] N. A. Dodgson, *Optical devices: 3D without the glasses*, Nature, **495**(7441), 316–317 (2013) [doi:10.1038/495316a].

[136] J. Sun, E. Timurdogan, A. Yaacobi, E. S. Hosseini, and M. R. Watts, *Large-scale nanophotonic phased array*, Nature, **493**(7431), 195–199 (2013) [doi:10.1038/nature11727].

[137] D. E. Smalley, Q. Y. J. Smithwick, V. M. Bove, J. Barabas, and S. Jolly,

Anisotropic leaky-mode modulator for holographic video displays, Nature, **498**(7454), 313–317 (2013) [doi:10.1038/nature12217].

[138] T. Xu, A. Agrawal, M. Abashin, K. J. Chau, and H. J. Lezec, *All-angle negative refraction and active flat lensing of ultraviolet light*, Nature, **497**(7450), 470–474 (2013) [doi:10.1038/nature12158].

[139] V. G. Veselago, *The Electrodynamics of substances with simultaneously negative values of ε and μ*, Sov. Phys. Usp., **10**(4), 509 (1968) [doi:10.1070/PU1968v010n04ABEH003699].

[140] J. Wallace, *Scientists build simple UV-focusing metamaterial flat lens*, Laser Focus World (2013).

[141] P. Valley, D. L. Mathine, M. R. Dodge, J. Schwiegerling, G. Peyman, and N. Peyghambarian, *Tunable-focus flat liquid-crystal diffractive lens*, Opt. Lett., **35**(3), 336–338 (2010) [doi:10.1364/OL.35.000336].

[142] H.-C. Lin, M.-S. Chen, and Y.-H. Lin, *A Review of Electrically Tunable Focusing Liquid Crystal Lenses*, Transactions on Electrical and Electronic Materials, **12**(6), 234–240 (2011) [doi:10.4313/TEEM.2011.12.6.234].

[143] R. Ng, *Digital light field photography*, Ph.D., Stanford University (2006).

[144] E. Buckley, *Computer-generated holograms for real-time image display and sensor applications*, Ph.D., University of Cambridge (2007).

[145] E. Hecht, *Optics*, Addison-Wesley (1998).

[146] T.-C. Poon, *Optical Scanning Holography with Matlab*, Springer (2007).

ACADEMIC REGISTRY
Research Thesis Submission

Name*:*	Javid Khan		
School/PGI:	Engineering and Physical Sciences, EPS		
Version: *(i.e. First, Resubmission, Final)*	Final	Degree Sought (Award **and** Subject area)	Eng.D. Photonics

Declaration

In accordance with the appropriate regulations I hereby submit my thesis and I declare that:

1) the thesis embodies the results of my own work and has been composed by myself
2) where appropriate, I have made acknowledgement of the work of others and have made reference to work carried out in collaboration with other persons
3) the thesis is the correct version of the thesis for submission and is the same version as any electronic versions submitted*.
4) my thesis for the award referred to, deposited in the Heriot-Watt University Library, should be made available for loan or photocopying and be available via the Institutional Repository, subject to such conditions as the Librarian may require
5) I understand that as a student of the University I am required to abide by the Regulations of the University and to conform to its discipline.

* *Please note that it is the responsibility of the candidate to ensure that the correct version of the thesis is submitted.*

Signature of Candidate*:*		Date:	

Submission

Submitted By *(name in capitals):*	Javid Khan
Signature of Individual Submitting:	
Date Submitted:	

For Completion in the Student Service Centre (SSC)

Received in the SSC by *(name in capitals):*			
Method of Submission *(Handed in to SSC; posted through internal/external mail):*			
*E-thesis Submitted (**mandatory for final theses)***			
Signature:		Date:	

Please note this form should bound into the submitted thesis.

Updated February 2008, November 2008, February 2009, January 2011